AÑO III - NUM. 60

MICROHOBBY

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR

SEMANAL

AÑO III - N.º 60

135 PTS.

(incluye IVA)

LOS OPERADORES LÓGICOS

SIR FRED. Un caballero Made in Spain

CÓMO REALIZAR RÓTULOS «DE PELÍCULA»

¡Feliz Año Nuevo!

GRABACION DE FICHEROS EN EL INTERFACE BETA

HOBBY PRESS, S.A.

Director Editorial
José I. Gómez-Centurión
Director Ejecutivo
Domingo Gómez
Asesor Editorial
Gabriel Nieto
Redactor Jefe
Africa Pérez Tolosa
Diseño
Rosa María Capitel
Redacción
Amalio Gómez, Pedro Pérez, Jesús Alonso
Secretaría Redacción
Carmen Santamaría
Colaboradores
Primitivo de Francisco, Rafael Prades, Miguel Sepúlveda Sergio Martínez y J. M. Lazo
Corresponsal en Londres
Alan Heap
Fotografía
Javier Martínez, Carlos Candel
Portada
José María Ponce
Dibujos
J. R. Ballesteros, A. Perera, F. L. Frontán, Pejo, J. M. López Moreno, J. Igual, J. A. Calvo, Lóriga, J. Olivares

Edita
HOBBY PRESS, S. A.
Presidente
María Andrino
Consejero Delegado
José I. Gómez-Centurión
Jefe de Publicidad
Marisa Esteban
Publicidad Barcelona
José Galán Cortés
Tels.: 303 10 22 - 313 71 76
Secretaría de Dirección
Marisa Cogorro
Suscripciones
M.ª Rosa González
M.ª del Mar Calzada
Redacción, Administración y Publicidad
La Granja, 39
Polígono Industrial de Alcobendas
Tel.: 654 32 11
Telex: 49480 HOPR
Dto. Circulación
Carlos Peropadre
Distribución
Coedis, S. A. Valencia, 245
Barcelona
Imprime
Rotedic, S. A. Ctra. de Irún, km. 12,450 (MADRID)
Fotocomposición
Espacio y Punto, S. A.
Paseo de la Castellana, 268
Fotomecánica
Grof
Ezequiel Solana, 16
Depósito Legal
M-36.598-1984

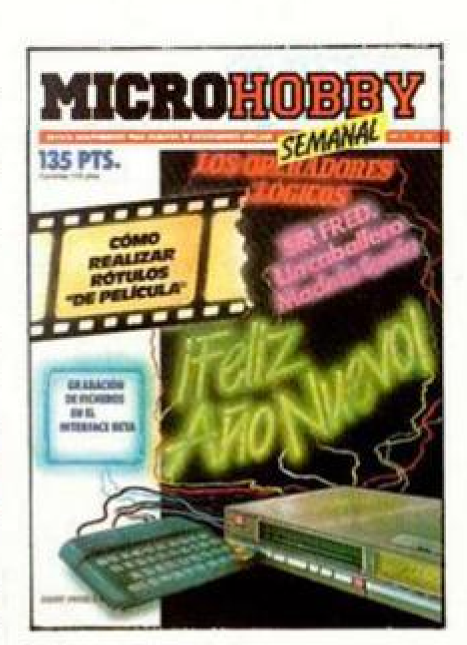
Representante para Argentina, Chile, Uruguay y Paraguay, Cía. Americana de Ediciones, S.R.L. Sud América 1.532. Tel.: 21 24 64. 1209 BUENOS AIRES (Argentina).

Solicitado control OJD

El precio será el mismo para Canarias que para la península y en él irá incluida la sobretasa aérea.

MICROHOBBY ESTA SEMANA

AÑO III. N.º 60. 7 al 13 de enero de 1986
135 ptas. (incluido I.V.A.)





Sir Fred, un caballero Made in Spain.
Pag. 12.

PREMIADOS CONCURSO «WEST BANK»

1. ALFREDO ARAUJO FERNANDEZ. Madrid.
2. FRANCISCO MARTINEZ NUÑO. Madrid.
3. DAVID BALLESTEROS GUILLEN. Madrid.
4. ANDRES ESPUÑAL FERRERO. Pamplona (Navarra).
5. IGNACIO DE LA ASUNCION. Burgos.
6. PABLO DE LA IGLESIA. Madrid.
7. ROBERTO SOBRINO SANCHEZ. Madrid.
8. RAUL CUEVAS BARRON. Sevilla.
9. DAVID CAMBLOR RUIZ. El Entrego (Asturias).
10. LOLY SANCHEZ FERRERO. La Coruña
11. JUAN I. MIEZA FERNANDEZ Madrid.
12. JOSE ALBERTO PEREZ GARCIA Lezo (Guipúzcoa).
13. OSCAR NICOLAS GOMEZ SANZ Pozuelo (Madrid).
14. ANTONIO MARTI SEVILLA Barcelona
15. FCO. JAVIER DE ARCE BALLESTEROS. Toledo
16. JOSE L. SANCHEZ. Madrid.
17. JOSE LUIS MERINO ESPAÑA. Santander.
18. FRANCISCO TEJEDA. Villalba (Madrid).
19. CARLOS ARANDA. Madrid.
20. JOSE MANUEL GARCIA JIMENEZ. Sevilla.
21. CARLOS BERMUDO MELERO. Zaragoza.
22. PEDRO JOSE SANCHEZ MARTINEZ. Cartagena (Murcia).
23. ANTONIO E. NAVAS GONZALEZ. Motril (Granada).
24. LUCIO GONZALEZ GONZALEZ. Vecindario (Gran Canarias).
25. BASILI SELLEZ MARTIN. Manresa (Barcelona).

PING-PONG DE IMAGINE

Parece que la casa de software Imagine tiene pensado llevar al ordenador la totalidad de los deportes existentes hasta el momento.

Tras los enormes éxitos obtenidos con programas anteriores como Hypersports o World Series Baseball, vuelve de nuevo a la brecha con un juego de simulación deportiva.



En esta ocasión le ha tocado el turno a un trepidante juego de mesa: el Ping-Pong, del cual podemos asegurar de antemano que alcanzará las más altas cotas de fama tan sólo por el hecho de llevar en la carátula el nombre impreso de Imagine.

Con él podremos sentir toda la emoción y tensión que una partida de Ping-Pong nos puede ofrecer, no sólo por el gran realismo y variedad de golpes que podremos asestar (mates, reveses, etc.), sino también por todo el calor y entusiasmo que un público espectante le puede ofrecer.

Como viene siendo habitual en este tipo de juegos, éste podrá ser utilizado por dos jugadores simultáneamente o por uno contra el ordenador.

Pin-Pong estará disponible en las diferentes versiones para Spectrum, Commodore 64 y Amstrad en un período no muy largo de tiempo.

MERECIA LA PENA ESPERAR

Como habréis podido comprobar el programa de Dinamic, Camelot Warrior, ha salido con unas cuantas semanas de retraso con respecto a la fecha prevista.

Esto ha sido debido a que en el último momento decidieron llevar a cabo una sustancial mejora en el programa, para lo cual tuvieron que hacernos esperar un poco más de la cuenta pero que una vez vistos los resultados obtenidos hemos comprobado que merecía la pena esperar.

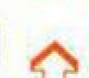
Esta mejora consistió en unas variaciones en el sistema de scroll de la pantalla y que representa una innovadora técnica de programación.

Lo que se ha hecho es que cuando el personaje principal entra en una nueva pantalla, ésta en lugar de permanecer estática durante todo el tiempo en la que permanezcamos en ella, va «scrolando» poco a poco de tal manera que los enemigos y peligros se van viniendo encima del personaje, por lo que apenas podemos perder tiempo en reaccionar y debemos ir tomando las decisiones de actuación de una manera rápida y continuada. El resultado es francamente brillante y demuestra el continuo afán de Dinamic por ir mejorándose día a día.

MICRO HITS


1 **Spy vs Spy**
(Beyond)

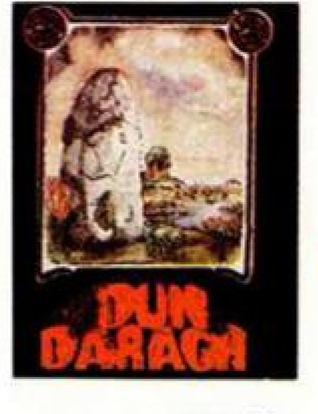

2 **Dum Darach**
(Gargoyle Games)


3 **Nodes of Yesod**
(Odin)


4 **Dinamite Dan**
(Mirrorsoft)


5 **World Series Basketball**
(Imagine)


6 **That's the Spirit**
(The Edge)



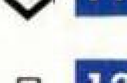

7 **Exploding Fist**
(Melbourne House)


8 **West Bank**
(Dinamic)


9 **Fighting Warrior**
(Melbourne House)

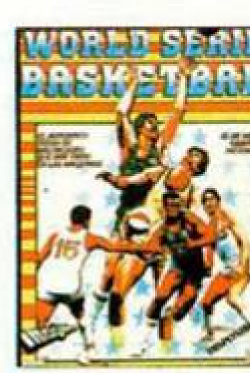

10 **Nightshade**
(Ultimate)

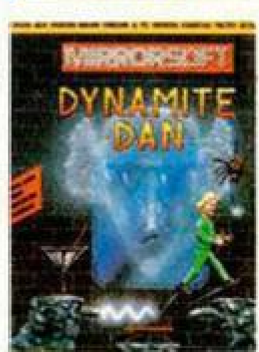

11 **Highway Encounter**
(Vortex)



12 **Glass**
(Quicksilva)


13 **Hypersports**
(Imagine)





COMANDO SE VA AL CINE



Por primera vez en la corta historia del software, un programa alcanza tanto éxito que una película es realizada basándose en su argumento. Hasta el momento lo normal era que ocurriera todo lo contrario, es decir, que algunos juegos intentaran buscar una fácil fama amparándose en el éxito previo de alguna conocida película.

Sin embargo, en Estados Unidos ha sido tal la conmoción que ha causado este Comando, tanto en el ámbito de los ordenadores personales como de las máquinas de juegos de los billares (prácticamente no existe ninguna sala que no lo posea), que el director de cine Mark L. Lester, decidió llevar a la pantalla grande un film que narrara las peripecias de este intrépido personaje.

La película, que ha sido estrenada en España en fechas recientes, está protagonizada nada más y nada menos que por el famoso actor y hercúleo culturista Arnold Schwarzenegger, conocido por los papeles representados en ambas partes de Connan y Terminator. No podían haber encontrado un actor más apropiado para un tan agresivo y monstruoso papel.

14 ***Fairlight*** *(The Edje)*

15 ***Supertest*** *(Ocean)*

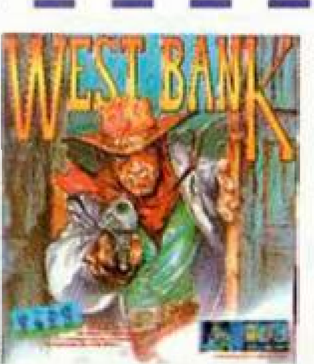
16 ***Spy Hunter*** *(U.S. Gold)*

17 ***Sourthern Belle*** *(Hewson Consultants)*

18 ***Juggernaut*** *(C.R.L.)*

19 ***Cauldron*** *(Place Software)*

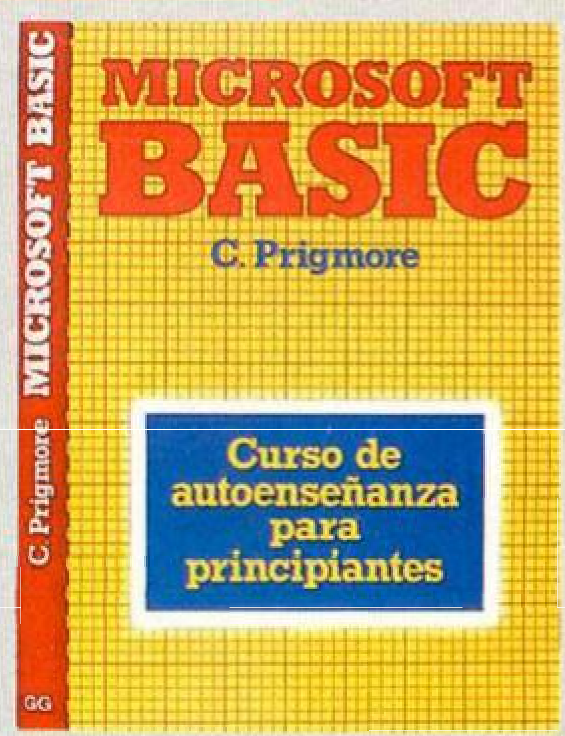
20 ***Profanation*** *(Dinamic)*







LIBROS



BASIC MICROSOFT, Curso de Autoenseñanza para principiantes

C. Prigmore. Ed. Gustavo Gili

Si hay algún tipo de libro que domina el mercado actualmente, éste es el de los que abordan temas básicos para principiantes. Sin embargo, pocos son los títulos que están estructurados como auténticos cursos de autoenseñanza, con niveles progresivos de dificultad y con ejemplos prácticos y didácitos donde se requieren.

Por otra parte, el Basic como lenguaje de alto nivel, tiene una enorme cantidad de dialectos, basados en cada una de las peculiaridades de los distintos modelos de Ordenadores. La versión de Basic más extendida y utilizada por mayor número de ordenadores, es el BASIC MICROSOFT, abordado por este libro.

Como reza en la presentación, está dirigido a enseñarnos a crear nuestros propios programas de entretenimiento, educación y para ayudarnos en las tareas domésticas y de pequeña empresa.

El libro presupone un total desconocimiento de este lenguaje de programación y hace hincapié, más en los métodos y procedimientos de programación que en los detalles técnicos concretos de cada ordenador, los cuales, se supone, deben encontrarse en los propios manuales de instrucciones.

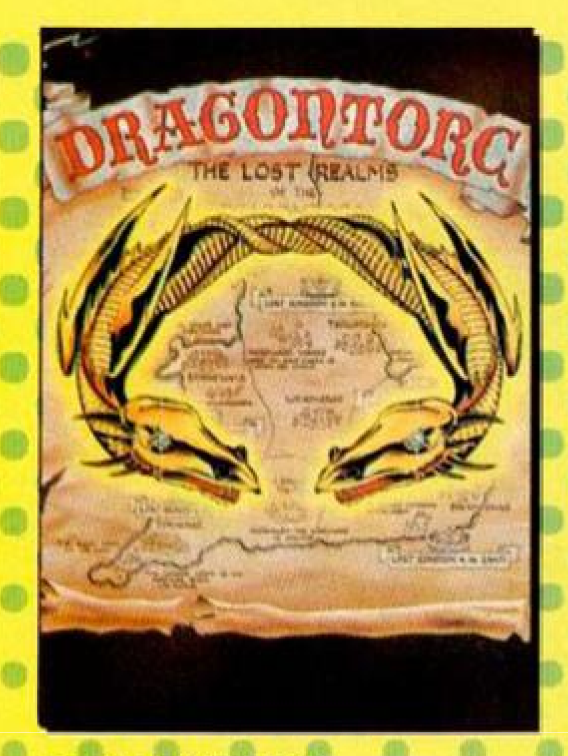
Un último detalle digno de mención es que para sacarle provecho no es necesario poseer ningún ordenador ya que los ejemplos muestran también la salida en pantalla prevista en cada uno de los casos.

¡NUEVO!

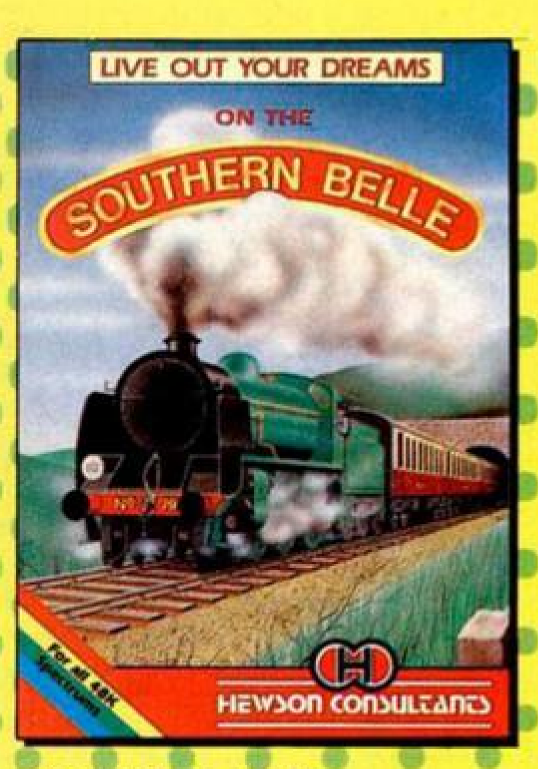
SIEMPRE LOS PRIMEROS EN TENER LO ULTIMO



P.° de la Castellana, 268, 3.° C. 28046-MADRID.
Tel.: (91) 733 25 00



RAID OVER MOSCOW
PLAY IT LIKE THERE'S NO TOMORROW!



DRAGONTORC

Cerca de 200 pantallas con miles de objetos diferentes y más de cien personajes con animación en tres dimensiones, hacen que de este juego la revista inglesa Crash Micro haya llegado a decir «Dragontorc es lo mejor que hemos visto en juegos de acción y aventura».

P.V.P.: 2.300 ptas. (Amstrad).
Precio Socios C. de Soft: 2.070 ptas.
P.V.P.: 1.900 ptas. (Spectrum).
Precio Socios C. de Soft: 1.750 ptas.

RAID OVER MOSCOW

Defiende a USA y Canadá del ataque nuclear que ha lanzado Rusia contra ellos. Con tu escuadrilla habrás de hacer un viaje lleno de peligros hasta llegar al mismísimo Kremlin y destruir las bases de lanzamiento soviéticas. Gráficos y acción sensacionales.

P.V.P.: 2.300 ptas. (Amstrad).
Precio Socios C. de Soft: 2.070 ptas.
P.V.P.: 2.100 ptas. (Spectrum).
Precio Socios C. de Soft: 1.890 ptas.

SOUTHERN BELLE

Siente la emoción de conducir una locomotora de vapor, através de un maravilloso recorrido desde Londres a Brigthon, manejando la caldera, el silbato, atravesando túneles, etc. Estamos ante uno de los juegos más brillantes y originales aparecidos para ordenador.

P.V.P.: 2.300 ptas. (Amstrad).
Precio Socios C. de Soft: 2.070 ptas.
P.V.P.: 2.100 ptas. (Spectrum).
Precio Socios C. de Soft: 1.890 ptas.

...Y LOS TRES PROGRAMAS POR SOLO 5.500 PTAS. (AMSTRAD) 4.950 PTAS. (SPECTRUM)

¡HAZTE HOY MISMO SOCIO DEL CIRCULO DE SOFT! Además de poder adquirir tus programas al mejor precio, recibirás información de forma periódica y gratuita, del mejor software que aparezca en el mercado.

¿QUE HAY QUE HACER PARA SER SOCIO DEL CIRCULO DE SOFT? Así de fácil: envíanos por correo tu nombre, dirección y modelo de ordenador, o bien, pide por teléfono o por correo tu primer programa. ¡Y entrarás a formar parte del CIRCULO DE SOFT de forma inmediata!

☐ SÍ, quiero ser SOCIO desde hoy mismo del CIRCULO DE SOFT y recibir periódicamente información de novedades de software, así como beneficiarme desde hoy mismo de los precios reducidos reservados a los SOCIOS y de sus Ofertas Especiales. El ser SOCIO no me obliga a compra alguna.

Si prefieres formalizar tu compra por teléfono puedes hacerlo llamando al (91) 733 25 00. ¡¡NO SE COBRAN LOS GASTOS DE ENVIO POR CORREO!!

TITULO	P.V.P.	ORDENADOR

☐ Contrarreembolso ☐ Giro Postal ☐ Talón adjunto a Microamigo, S.A. ☐ Tarjeta VISA n.° ______ Fecha caducidad ______

Nombre ______ Apellidos ______ Edad ______

Domicilio ______ Teléfono ______

Localidad ______ C.P. ______ Provincia ______



COMO UNA BOMBA

Las posibilidades del Spectrum pueden ser infinitas si nos empeñamos en ello y para demostrarlo, aquí os ofrecemos un truco con el que conseguiréis la explosión de una auténtica bomba.

Probarlo y veréis...

```
10 FOR n=64647 TO 64674
20 READ a: POKE n,a: NEXT n
30 DATA 17,98,100,38,50,58,72,
92,31,31,31,14,254,238
40 DATA 16,237,121,67,16,254,3
7,32,244,28,21,32,232,201
```



CON TODA PRECISION

Este truco de Juan García Martínez te permitirá saber, con toda precisión de cálculo, el tiempo (en segundos) que el ordenador lleva encendido.

Una sencilla rutina que os evitará algún que otro susto de calentamiento.

```
10 PRINT (PEEK 23672+256*PEEK
23673+65536*PEEK 23674)/50
```

¿CUANTA MEMORIA?

Para averiguar la memoria libre de que disponemos en el Spectrum, tanto de 16 como de 48K, podemos llamar a una subrutina de la ROM, que hace precisamente eso y lo hace de la siguiente forma, tanto en una línea BASIC como en comando directo: PRINT 65535 USR 7962.

Otra idea, también de Rafael Moragrega, ésta para saber el espacio que ocupa el programa BASIC, consiste en averiguar las direcciones PROG y E LINE de las variables del sistema y luego, restar la primera de la segunda, lo que puede hacerse de la siguiente forma:

```
10 LET l=23642
20 PRINT (PEEK l*256+PEEK (l-1
))-(PEEK (l-6)*256+PEEK (l-7))
```

FIREFOX

TODAS LAS LETRAS MAYUSCULAS SUBRAYADAS DEBERAN TECLEARSE EN MODO GRAFICO.

Vicente BLAZQUEZ

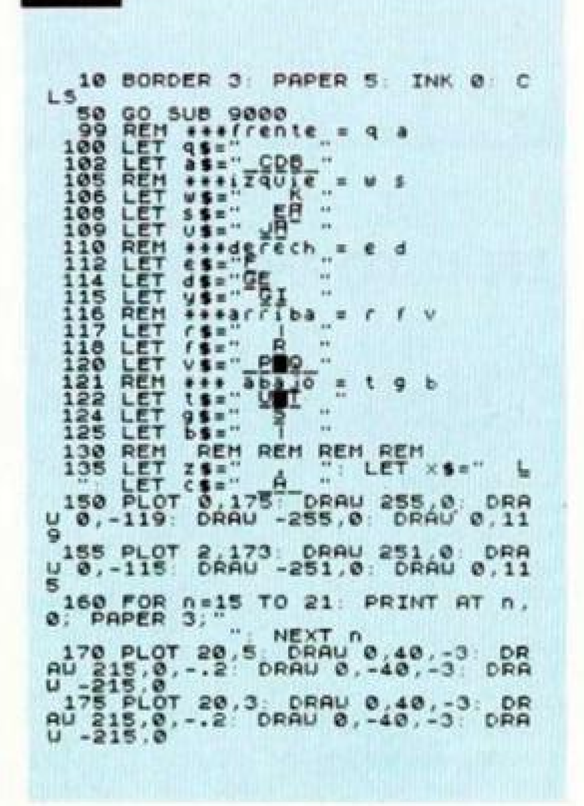
Spectrum 48 K

Un ultramoderno avión de combate ha sido creado a imagen y semejanza al de las fuerzas U.S.A., por los rusos. Nosotros, expertos pilotos a sueldo, hemos de hacer lo imposible por acabar con él.

La misión, nada fácil si tenemos en cuenta las características de la nave a destruir, tiene que llevarse a cabo en el menor tiempo posible para lo que tenemos que contar con la habilidad que nos caracteriza y el acoplamiento total con el aparato.

Siempre atentos al panel de mandos controlaremos la situación y llevaremos, estamos seguros, nuestro objetivo a buen fin.

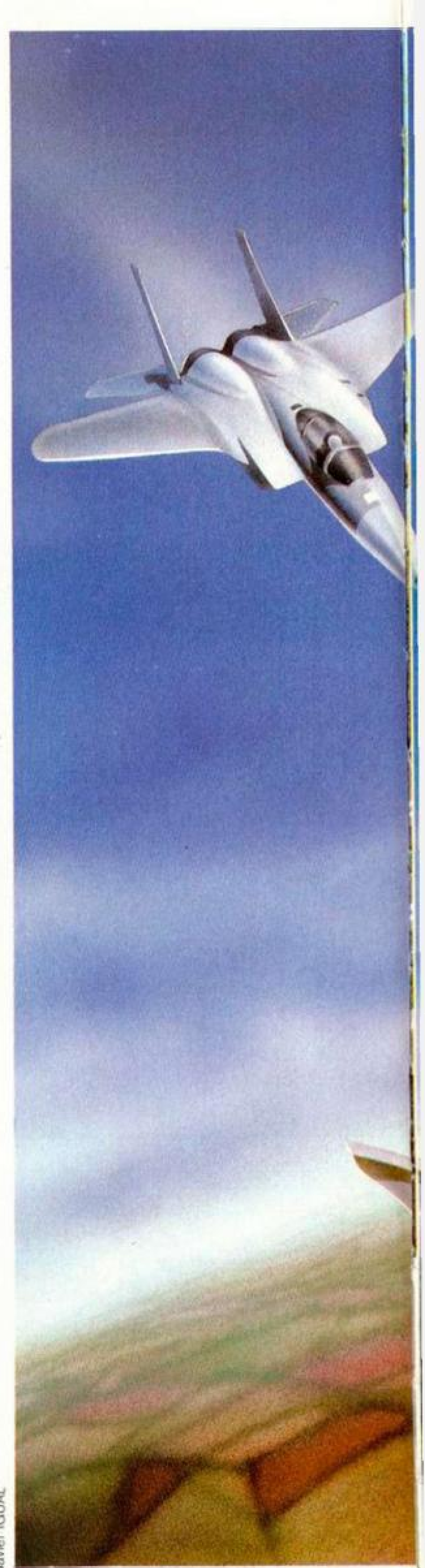




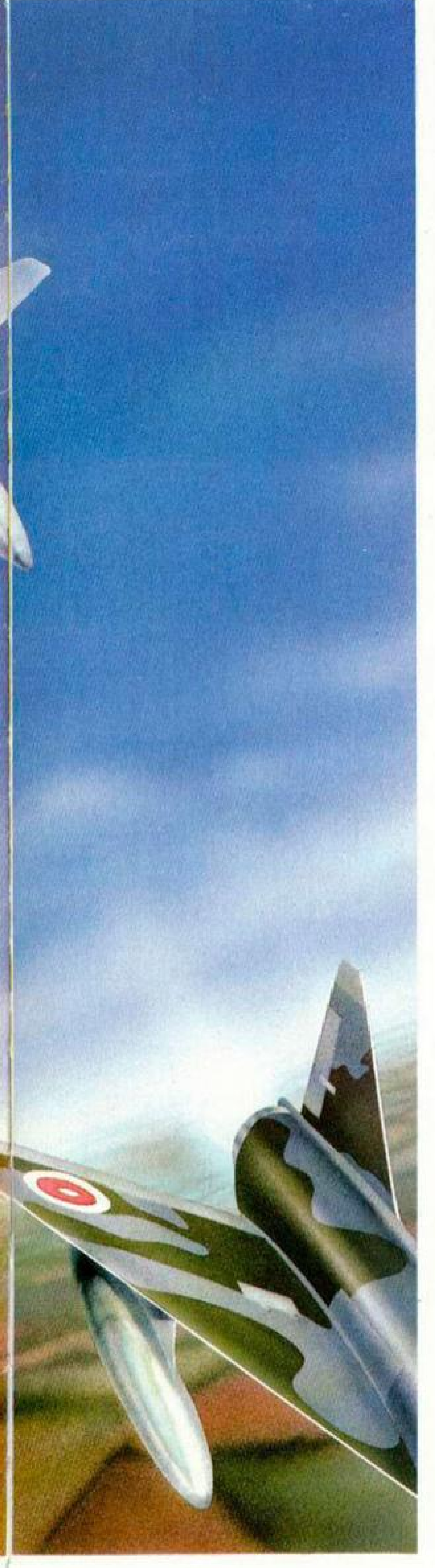
```
  10 BORDER 3: PAPER 5: INK 0: CLS
  50 GO SUB 9000
  99 REM ***frente = q a
 100 LET q$="    "
 102 LET a$=" CDB "
 105 REM ***izquie = w s
 106 LET w$="   K "
 108 LET s$="  EA "
 109 LET u$=" JA  "
 110 REM ***derech = e d
 112 LET e$="F    "
 114 LET d$="GE   "
 115 LET y$=" GI  "
 116 REM ***arriba = r f v
 117 LET r$="  I  "
 118 LET f$="  R  "
 120 LET v$=" P Q "
 121 REM *** abajo = t g b
 122 LET t$=" U I "
 124 LET g$="  S  "
 125 LET b$="  T  "
 130 REM  REM REM REM REM
 135 LET z$="  A  ": LET x$="  L ": LET c$=" A "
 150 PLOT 0,175: DRAW 255,0: DRAW 0,-119: DRAW -255,0: DRAW 0,119
 155 PLOT 2,173: DRAW 251,0: DRAW 0,-115: DRAW -251,0: DRAW 0,115
 160 FOR n=15 TO 21: PRINT AT n,0; PAPER 3;"                                ": NEXT n
 170 PLOT 20,5: DRAW 0,40,-3: DRAW 215,0,-.2: DRAW 0,-40,-3: DRAW -215,0
 175 PLOT 20,3: DRAW 0,40,-3: DRAW 215,0,-.2: DRAW 0,-40,-3: DRAW -215,0
```



```
 180 PLOT 30,25: DRAW 35,0: DRAW 0,-19: DRAW -35,0: DRAW 0,19
 185 PLOT 94,25: DRAW 42,0: DRAW 0,-19: DRAW -42,0: DRAW 0,19
 190 PLOT 158,25: DRAW 35,0: DRAW 0,-19: DRAW -35,0: DRAW 0,19
 195 PLOT 119,40: DRAW 49,0: DRAW 0,-9: DRAW -49,0: DRAW 0,9: PLOT 207,25: DRAW 35,0: DRAW 0,-11: DRAW -35,0: DRAW 0,11
 200 LET canti=0: LET imter=0: LET meda=15: LET color=0: LET puntos=0: LET km=200: LET nu=5: LET ve=5: LET fuel=299
 300 PRINT AT 17,15;"      ";AT 20,4;"     ";AT 20,12;"     ";AT 20,20;"     ";AT 19,26;"    "
 500 PRINT AT 3,3; INK 0;"        ";AT 4,3;"        ";AT 5,3;"        ";AT 6,3;"        ";AT 7,3;"        ";AT 8,3;"        "
 520 PRINT AT 9,3; INK 0;" CDB >>>>>   "FIRE FOX"";AT 10,3;z$;" ..... 300";AT 11,3;x$;" ..... 150     aprox.";AT 12,3;c$;" ..... 75"
 550 FOR n=1 TO 13: PRINT AT n,1; OVER 1; PAPER 2;"                              ": BEEP .01,n: NEXT n
 560 FOR n=13 TO 1 STEP -1: PRINT AT n,1; OVER 1; PAPER 5;"                              ": BEEP .01,n: NEXT n
 580 PRINT AT 13,13; FLASH 1;"PULSE": IF INKEY$<>"" THEN BEEP .9,10: BEEP .5,20: GO TO 700
```



Javier IGUAL






```
 600 BEEP .9,-10: BEEP .5,0: GO TO 550
 700 FOR n=1 TO 13: PRINT AT n,1; PAPER 5;"                    ": NEXT n
 900 LET o$=q$: LET k$=a$: LET m$=""
 910 LET nume=0: LET ene=10-INT (RND*9): LET mi=25-INT (RND*24): LET n$=z$
1000 PRINT AT nu,ve; PAPER 5; INK 7;"      ";AT nu+1,ve;"      ";AT nu+2,ve;"      ";AT nu+3,ve;"      "
1100 PRINT AT ene,mi;n$
1200 PRINT AT 5,15; OVER 1; BRIGHT 1;" "
1210 PRINT AT 17,24; OVER imter; INK 0; PAPER 3;"O"
1400 LET nume=nume+1: LET fuel=fuel-1
1410 IF nume=3 THEN LET n$=x$
1415 IF nume=7 THEN LET n$=c$
1420 IF nume=13 THEN PRINT AT ene,mi;"  ": GO TO 910
1500 PRINT AT 10,13;o$
1502 PRINT AT 11,13;k$
1504 PRINT AT 12,13;m$
1530 LET o$=q$: LET k$=a$: LET m$="    "
1540 IF fuel<=0 THEN GO TO 3000
1550 PRINT AT 19,4;"FUEL";AT 20,4;fuel;" "
1560 PRINT AT 19;12;" ALT.";AT 20,12;INT km;" "
1570 PRINT AT 19,20;"SCOR";AT 20,20;puntos
1580 PRINT AT 19,26;"M-";canti
1600 IF INKEY$="q" OR INKEY$="7" THEN LET km=km+20: LET o$=r$: LET k$=f$: LET m$=v$: LET nu=nu+.1: PRINT AT ene,mi;"  ": LET ene=ene+1
1605 IF INKEY$="a" OR INKEY$="6" THEN LET km=km-20: LET o$=t$: LET k$=g$: LET m$=b$: LET nu=nu-.1: PRINT AT ene,mi;"  ": LET ene=ene-1: PLOT 104,95: DRAW 38,0
1610 IF INKEY$="o" OR INKEY$="5" THEN LET o$=w$: LET k$=s$: LET m$=u$: LET ve=ve+.15: LET mi=mi+1
1615 IF INKEY$="p" OR INKEY$="8" THEN LET o$=e$: LET k$=d$: LET m$=y$: LET ve=ve-.15: LET mi=mi-1
1620 IF INKEY$="m" OR INKEY$="0" THEN LET fuel=fuel-5: PLOT 121,90: DRAW 2,40: PLOT 126,90: DRAW -2,40: FOR n=10 TO 5 STEP -1: OUT 255,10: BEEP .01,-10: PRINT AT n,15; PAPER 5;" ": OUT 200,100: NEXT n: BORDER 3: IF ene=5 AND mi=13 THEN GO TO 2010
1700 IF nu>=10 THEN FOR n=nu TO nu+3: PRINT AT n,ve;"      ": NEXT n: LET nu=2
1705 IF nu<=1 THEN FOR n=nu TO nu+3: PRINT AT n,ve;"      ": NEXT n: LET nu=9
1710 IF ve<=1 THEN FOR n=nu TO nu+3: PRINT AT n,ve;"      ": NEXT n: LET ve=22
1715 IF ve>=23 THEN FOR n=nu TO nu+3: PRINT AT n,ve;"      ": NEXT n: LET ve=2
1720 IF ene>=13 OR ene<=1 OR mi<=1 OR mi>=25 THEN PRINT AT ene,mi;"  ": GO TO 900
1725 IF ene=5 AND mi=13 THEN LET imter=1
1730 IF ene<>5 AND mi<>13 THEN LET imter=0
1735 IF km<=20 OR km>=1000 THEN GO TO 3000
2000 GO TO 1000
2005 REM DI S P A RO
2010 IF n$=z$ THEN LET xz=300
2015 IF n$=x$ THEN LET xz=150
2020 IF n$=c$ THEN LET xz=75
2050 FOR n=puntos TO puntos+xz STEP 5: PRINT AT 20,20;n: BEEP .01,30: NEXT n: LET puntos=n
2060 PRINT AT 5,14;"  "
2080 PRINT AT 17,5; INK 0; PAPER 3;"HONOR :  ";AT 17,meda; INK color; PAPER 7;"N": LET meda=meda+1: LET color=color+1
2090 IF color=6 THEN BEEP .5,0: LET puntos=puntos+300: LET meda=15: PRINT AT 17,15;"      ": LET color=0: LET fuel=fuel+50: LET canti=canti+1
2100 GO TO 910
3000 REM  THE END
3010 PRINT AT 11,13;"\F OG";AT 12,13
3020 PRINT AT 12,1; INK 2;" H A P Q    HH A  ";AT 13,4;" "
3050 FOR n=0 TO 50: BEEP .001,RND*30: PLOT 155-(RND*60),95-(RND*25): PRINT AT 0,9; INK RND*6; BRIGHT 1;" T H E  E N D ": NEXT n: INK 0
3055 PRINT AT 5,4;"PUNTOS:";AT 6,4;puntos
3060 PRINT AT 2,11; BRIGHT 1; INK 0; OVER 1;"O T R A ? ": OVER 0: BEEP .2,-30
3065 IF INKEY$="s" THEN BEEP .1,1: RUN
3070 IF INKEY$="n" THEN STOP
3080 GO TO 3060
4999 STOP
8999 BEEP 10,1: STOP
9000 REM G.DEFINIDOS
9010 FOR g=USR "a" TO USR "u"+7
9015 READ dato
9020 POKE g,dato
9025 NEXT g
9100 DATA 0,0,24,60,36,60,255,255
9105 DATA 0,0,0,0,0,225,163,255
9110 DATA 0,0,0,0,0,135,197,255
9115 DATA 52,118,126,102,219,219,231,153
9120 DATA 157,252,203,183,119,110,158,111
9125 DATA 128,192,96,112,120,60,62,25
9130 DATA 15,7,3,1,0,0,0,0
9135 DATA 240,224,192,128,0,0,0,0
9140 DATA 128,192,32,32,240,120,28,6
9145 DATA 1,3,4,12,31,62,120,192
9150 DATA 1,2,6,14,28,60,88,152
9155 DATA 0,0,0,24,24,255,0,0
9160 DATA 56,16,16,56,56,124,124,56
9165 DATA 126,60,24,60,36,102,126,60
9170 DATA 20,44,94,131,151,110,60,0
9175 DATA 0,0,0,0,1,79,255,191
9180 DATA 0,0,0,0,128,242,255,253
9185 DATA 24,24,36,102,255,231,219,165
9190 DATA 36,36,60,126,126,189,24,24
9195 DATA 125,255,250,224,128,0,0,0
9200 DATA 190,255,95,7,1,0,0,0
9980 RETURN
9999 SAVE "FIRE FOX" LINE 1: BEEP 6,30
```

DE LA CONTINUIDAD A LA FAMA

Carmen SANTAMARIA

Paco, Charli, Fernando y Camilo, los famosos chicos del Fred, asoman de nuevo a nuestras páginas tras la inminente aparición de su última creación, Sir Fred, basado en el mismo personaje que les lanzara, allá por el pasado año, a la popularidad en el mundo de la programación. Este, principalmente, es el motivo de esta entrevista a los componentes de Made in Spain.

¿Cómo surgió la idea de hacer un nuevo programa con este mismo personaje?

• En realidad, la primera idea fue la de hacer un programa, y en base a esto, después de terminarlo surgió el mismo personaje, debido a que el dibujo se parecía al anterior juego.

¿Cuál ha sido el método de trabajo, tanto en la idea como en los gráficos?

• Trabajamos con tres ordenadores (Spectrum 48K), el ordenador principal con la unidad de disco, era donde se escribía realmente el programa y los otros dos tenían una unidad de Microdrive respectivamente. En uno de éstos se probaba el programa y el otro lo utilizábamos para los gráficos. De las cuatro personas que formamos el grupo, «Charlie», sólo se dedicaba a hacer los gráficos, pero también participaba en llevar a cabo el programa, y los demás exclusivamente hacíamos el juego.



Paco Menéndez.

La idea es bastante aceptable y estamos muy contentos, pudiendo decir que es el n.º 1 de los realizados hasta ahora por nosotros.

En cuanto a los gráficos, la idea fundamental al empezar el programa fue la de cómo codificar las pantallas, después pensamos que fuera en un castillo y por supuesto, pusimos todo nuestro empeño en cuidar el movimiento de nuestro personaje.

¿Cuánto tiempo empleasteis para llevarlo a cabo?

• A primeros de octubre del año 1984, empezamos a programarlo y lo terminamos un año después, es decir en octubre de 1985, aunque después tuvimos que retocarlo antes de lanzarlo como exclusiva. Más o menos nos llevaba una media de tres horas al día, y después durante el verano, las ocho horas.

¿Qué problemas encontráis en la distribución?





Charlie Granados.

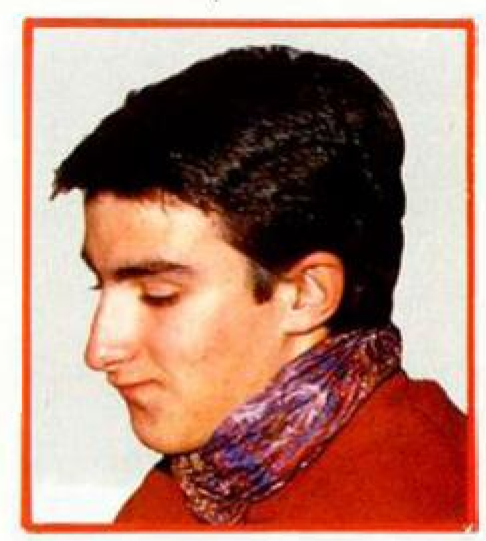
• Respecto a España, tenemos que señalar que Pablo de la casa Dinamic, nos ha ayudado bastante. De todas formas el Corte Inglés nos ha pedido 1.000 copias y adelantamos que ya están a su disposición. Todavía no disponemos de ninguna exclusiva para la distribución en España, pero no perdemos las esperanzas.

¿Qué os llevó a elegir a Mikro-Gen, como distribuidora en Inglaterra?

• A finales de septiembre, grabamos un vídeo y lo enviamos a doce casas inglesas, respondiéndonos ocho. Mantuvimos una correspondencia con éstas y cuando fuimos a Inglaterra (primeros de Octubre) debido a las ofertas recibidas, seleccionamos cuatro casas: Firebird, Elite, Mikro-Gen y Ultimate. Debido a dificultades de tiempo sólo pudimos contactar con Mikro-Gen y Elite. Nuestra idea inicial para la distribución era Elite, pero la oferta era un tanto pobre, a pesar de que en un principio fue similar a la propuesta por Mikro-Gen. Por eso elegimos esta casa.



Camilo Cela.



Fernando Rada.

¿Estáis satisfechos, en general, de como ha quedado el programa?

Reconocemos que quizá la pantalla de presentación no es tan brillante como se esperaba, pero la memoria no daba para más, incluso tuvimos que rechazar algunas cosas, pero no creemos que sea difícil de manejar, y por supuesto estamos bastante satisfechos y convencidos de que será un buen programa.

¿Qué opináis de la programación que se está haciendo en España?

Comparada con Inglaterra la programación en España es escasa, aunque hay juegos y personas que llegan al nivel alcanzado en Inglaterra.

Referente a lo económico ¿merece la pena hacer un programa?

Realmente está bien pagado, a pesar del tiempo empleado, ya que a parte de lo que nos ofrece la distribuidora, al pasarlo a otras versiones, también sacamos beneficio.

Una última pregunta ¿qué proyectos tenéis para un futuro?

No hay ninguna idea concreta ni segura, aunque no nos falta trabajo, puesto que estamos terminando la adaptación para la versión de Amstrad, y después descansaremos para, aproximadamente en febrero, volver a programar algún juego más.

SIR FRED • Arcade • Made in Spain

UN CABALLERO MADE IN SPAIN

Los autores de Fred, vuelven con renovadas aventuras para su simpático y narigudo personaje: Sir Fred. Pero esta vez le han puesto las cosas mucho más difíciles y los peligros que le aguardan son innumerables. Menos mal que sus recursos son inagotables.

Por fin parece que, poco a poco, los creadores de software españoles están comenzando a despertar de un largo letargo, que los buenos programadores están dejando de ser unos cuantos casos aislados y que, definitivamente, se están empezando a crear buenos productos. Evidentemente tampoco hay que echar las campanas al vuelo, pues está claro que la producción de juegos en nuestro país resulta, desde todo punto de vista, ridícula y escasísima, si la comparamos con la de otros países europeos.

Sin embargo, parece que las cosas empiezan a funcionar. Y un buen ejemplo lo tenemos en el caso de Made in Spain (simpático nombre), que tras el éxito obtenido con su anterior programa, Fred, vuelven a acaparar la atención de todos con una excelente creación: Sir Fred.

Aquélla su primera obra fue, sin duda alguna, uno de los mayores éxitos del software español; pero por otro lado, también se veía que era eso, el primer juego que realizaban, y se podía apreciar en él una cierta falta de madurez. Pero ahora todo esto ha quedado atrás. Con Sir Fred, se nota perfectamente la experiencia adquirida y un mayor dominio de las técnicas de programación. Evidentemente se ha realizado un programa que supera ampliamente al anterior.

Aquellos jóvenes y tímidos «muchachos del Fred», se han convertido en los señores de Made in Spain.

Pero dejemos a un lado todas estas cuestiones metafísicas y transcendentales y centrémonos en el tema que realmente nos interesa, en Sir Fred.

Y las primeras cuestiones surgen por sí mismas: ¿quién es Sir Fred? ¿qué es lo que pretende? Pues Sir Fred es un auténtico caballero, de ésos que ya se ven pocos por el mundo. A pesar de que su físico no le acompaña demasiado, pues es gordito, calvo y posee un nada despreciable apéndice nasal, sus virtudes como persona son verdaderamente excelentes: valeroso, aguerrido, incansable defensor del bien, luchador sin par y gallardo como pocos. En cuanto al asunto que le trae por aquí, es fácil de adivinar. ¿Qué es lo que puede mover a un caballero a realizar las más osadas proezas y embarcarse en las más peligrosas aventuras? Pues evidentemente el amor de una dama.

En cuanto llegó a sus oídos que su bella amada había sido raptada y que se encontraba encerrada en el castillo de los malvados Beni-Gómez, corrió sin perder un instante, y sin más armas que su ira y su arrojo, a intentar un arriesgado rescate.

Pero el castillo no es precisamente un convento de hermanitas de la caridad. En él se encierran todos los peligros que se puedan imaginar y algunos más. No, desde luego que no le va a resultar nada fácil entrar en el lugar

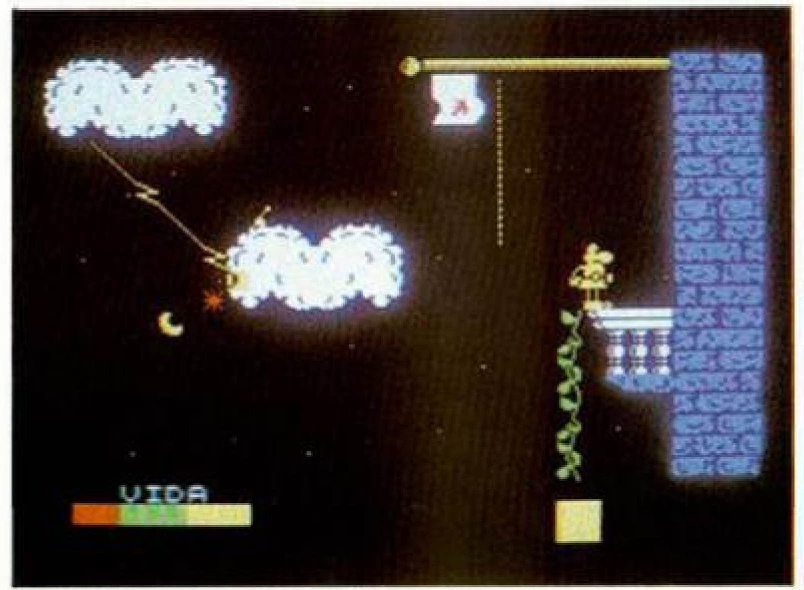


donde se encuentra su amada. Para ello deberá realizar todo tipo de peripecias y enfrentarse a un sin fin de animales, trampas y enemigos.

¿Y qué es lo que puede hacer para evitar estos inconvenientes?, os estaréis preguntando en estos momentos. Pues de todo, muchachos, de todo: caminar, saltar, escalar por las enrredaderas, balancearse en cuerdas y lianas, pelear con espadas, nadar, disparar flechas, y un gran número de acciones diferentes. Pero sobre todo, lo que más falta le hará será utilizar todo su ingenio y astucia para lograr que los objetos que se vaya encontrando por el castillo (sogas, espadas, arcos, botellas, cerillas, etc.), le sirvan de alguna utilidad y poder ir abriendo nuevos caminos para llegar a tan anhelado fin.

Pero como ya os hemos dicho, esta tarea le va a resular extremadamente





complicada y quizás éste sea uno de los pocos defectos que le encontramos al juego: el hecho de que para conseguir ir pasando de pantalla tienes que calentarte demasiado la cabeza; aunque posiblemente este sea uno de los detalles que hagan que cojamos un mayor interés por él, (sobre gustos...).

Sin embargo, lo más atrayente del juego es, sin duda, sus gráficos. Estos están realizados con una auténtica vocación artística y cada una de las pantallas está llena de figuras y colores, que nos sitúan en escenarios completamente diferentes, por lo que el resultado es, cuando menos, brillante en la mayoría de las ocasiones.

Y de los movimientos, pues qué os vamos a decir. Si los gráficos son buenos, éstos son aún mejor. A parte de la gran cantidad de acciones que puede llevar a cabo y que anteriormente os hemos comentado, éstos están hechos, entre otras cosas, con realismo y con humor. Por tanto, ésta es una nueva virtud que tenemos que añadirle al resultado general del programa.

Aunque justo es decir, que no hay nada perfecto, pues tanto lujo de detalles en la pantalla se paga con una presentación un tando pobre, pues los accescrios e indicativos de la vida y de los objetos llevados quedan algo sosos en relación al resto de la pantalla. Pero bueno, esta es una circunstancia que no tiene ninguna transcendecia y que puede ser totalmente pasada por alto.

En definitiva, Sir Fred es un juego muy bueno en todos sus aspectos y esperamos que los señores de Made in Spain se vean animados con esta creación a seguir realizándo nuevos programas, al menos tan simpáticos y entretenidos como éste.

STRONGMAN • Martech • Arcade

EL PODER DE LA FUERZA



Nos encontramos ante un juego original, cosa que por lo menos ya es bastante en los tiempos que corremos, pero que quizás desilusiona un poco.

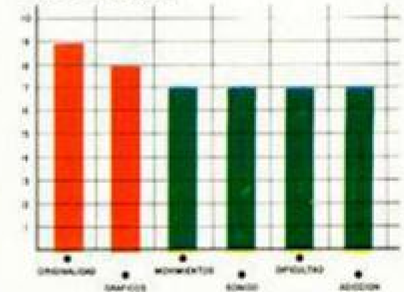
Eso de sentirte un fortachón e intentar igualar las proezas de Geoff Capes (Campeón mundial y europeo de Forzudos o como se diga), es, a priori, bastante atrayente, pero a la hora de la verdad el resultado es algo pobre en su conjunto.

Sin embargo, tampoco hay que exagerar. No es tan malo, todo lo contrario, lo que pasa es que como te esperas más te sientes un poco defraudado.

El juego consiste en ir superando las diferentes pruebas que se nos van presentando y para las cuales necesitaremos emplear distintos tipos de estrategia. Para algunas de estas pruebas, como por ejemplo el arrastrar un camión, o levantar en vilo un coche, lo que tendremos que hacer es distribuir adecuadamente la fuerza de la que dispongamos entre todos los músculos de nuestro cuerpo y una vez hecho ésto, es cuando tenemos que demostrar nuestros reflejos para ir utilizando cada uno de estos músculos en el momento exacto en el que lo necesitemos para poder acabar la prueba. Sin embargo, en otras ocasiones lo que deberemos hacer es poner al máximo nuestra puntería y pulso firme si queremos conseguir salir con éxito de pruebas como la de cortar troncos.

Lo bueno de este programa es que debido a la variedad de movimientos y acciones diferentes que debemos realizar no se hace demasiado pesado, pero de todas formas, ya os decimos que tampoco es una cosa del otro mundo. En fin, los hemos visto peores y que han gustado, por tanto no hay ninguna razón para que Strongman no tenga un cierto éxito.







THUNDERBIRDS • Firebird • Arcade

OPERACION RESCATE

Un grupo de Egiptólogos han quedado atrapados en el interior de una antigua pirámide. Por culpa de la complicación del laberinto que da acceso a la salida, han requerido la ayuda del exterior para que con dos naves especiales y un equipo adecuado, consigan llegar hasta el interior de la pirámide y rescatarles.

Pero debido al gran número de trampas y piedras deslizantes que caen para entorpecer el camino, los pilotos de las naves necesitarán utilizar al máximo toda su astucia y pericia para conseguir que los caminos se vayan abriendo ante ellos.

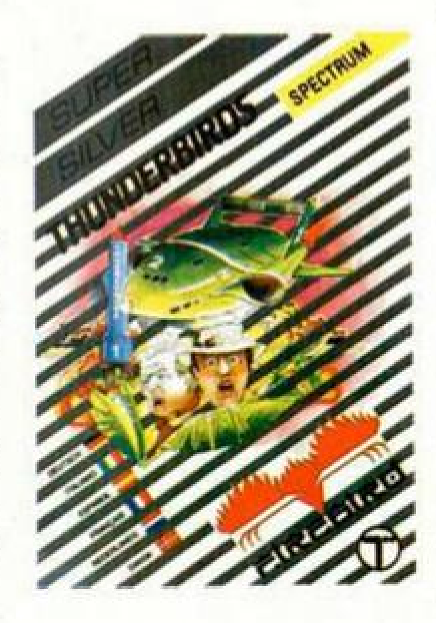
Este es el argumento de este Thunderbirds, pero como en la mayoría de los juegos, es tan solo un pretexto para darle algo de emoción y realismo a las diferentes acciones que debemos realizar en el ordenador. En esta ocasión lo que deberemos hacer es plantear en cada una de las pantallas la estrategia necesaria para ir pasando con ambas naves, puesto que con una sola sería imposible ya que se necesitan la una a la otra para

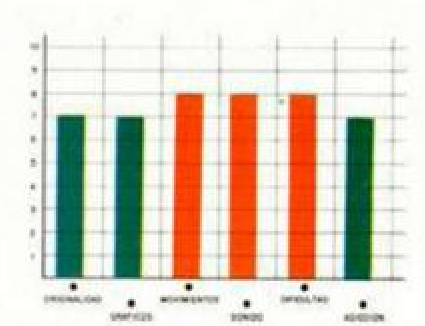


ir desplazando las rocas y poder seguir adelante.

El juego, en general, está bastante bien realizado y resulta entretenido, pero el mayor fallo que le vemos (y por cierto, esta es una circunstancia que estamos empezando a ver en demasiados juegos) es que antes del comienzo del mismo debes elegir el equipamento que quires llevar, y si no eliges el adecuado ya nada podrás hacer para completar la misión, por lo que deberás empezar de nuevo y seguir probando, con muchísima paciencia, hasta conseguir la combinación de objetos adecuada para conseguir acabar.

Esto desde luego resulta un poco tedioso, y es una pena porque el desarrollo del juego en sí es muy interesante, pues debes poner a prueba tu inteligencia y capacidad de estrategia.





ARABIAN NIGHTS • Interceptors Micros • Arcade

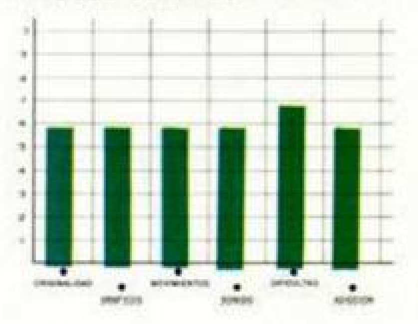
UN EXOTICO JUEGO

Las mil y una noches nos llega de la mano de una apenas conocida casa de software, Interceptors Micros. El juego tiene lugar en algunos de los innumerables escenarios exóticos en los que se desarrollan los cuentos que la bella Zoraida tuvo que inventarse para poder salvar su vida, (para mayor información consultar el libro original). Por ellos deberemos movernos con el príncipe de Kalendar, Imrahil, hasta conseguir, nada más y nada menos, que rescatar a una princesa, (creo que este argumento me suena de algo).

Y, ¿qué es lo que debe de hacer? Pues lo de costumbre en estos casos: salvar obstáculos, saltar de un lado a otro, coger cosas, y todo eso que se hace en un arcade típico.

Los gráficos y los movimientos tampoco son nada del otro mundo, pero al menos los primeros son variados y los segundos están bien realizados.

Pero bueno, de todas formas este Arabian Nights es un juego entretenido con el que podrás pasar unos ratos graciosos llevando a Imrahil por los distintos lugares paradisíacos y los maravillosos castillos orientales.



MICROHOBBY

AMSTRAD Semanal

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

LE OFRECE AHORA SUS PROGRAMAS YA GRABADOS PARA QUE VD. NO TENGA QUE TECLEARLOS. TOTALMENTE DESPROTEGIDOS CON EL OBJETO DE FACILITAR SU COPIA EN DISCO.

Todos los programadores y aficionados a la microinformática, sabemos lo tedioso y propenso a errores que resulta el teclear un listado de un programa. Para facilitar tu labor al máximo y evitar que malgastes largas horas sobre el teclado de tu ordenador tratando de descifrar incomprensiblemente mensajes de error.

AMSTRAD SEMANAL te ofrece cada mes los programas publicados en los cuatro números correspondientes, en una cinta de cassette desprotegida, que te permitirá copiar los programas en disco y tener acceso a los listados para su estudio y posterior edición de rutinas.

Programas incluidos en la cinta número 1

Título	Rev. n.	Título	Rev. n.
EASYDRAW	1	MAD ADDER	3
EGGBLITZ	2	HEXER	3
CODIGO SECRETO	2	CHARGEN	4
VENTANAS	2	PROGRAMACION	4
BIORRITMOS	3		

Programas incluidos en la cinta número 2

Título	Rev. n.	Título	Rev. n.
GRAFICOS	8	INCOGNITON	7
MUSICA	6	MONITOR	5
TRON	6	ANALISIS	5-8
ENSAMBLADOR	8	CEDRIC	8
HEXERL	8	ANIMACION1	7
TOOLKIT	8	ANIMACION2	8
PRIMEROS PASOS	7	SMALEY	7

Programas incluidos en la cinta número 3

Título	Rev. n.	Título	Rev. n.
ANALISIS	9-12	SPRITE112	11
FRUTAS	9	AMSCARD	11
MENUDISC	9	OTELO	12
RS×101	10	EVENT121	12
RS×102	10	EVENT122	12
CUATRORAYA	10	FRUTTIES	12
SPRITE111	11		

Por sólo 675 pts.
(incluidos gastos de envío)

COMPATIBLES CON LOS MODELOS CPC-464, CPC-664 y CPC-6128

Recíbelos en tu casa cómodamente enviándonos con la menor demora posible, el cupón que se encuentra en la última página de la revista

Un número de cuatro dígitos necesitaría dos octetos, etc. Ejemplo, el número 3427 se definiría:

0 0 1 1 0 1 0 0	34h
0 0 1 0 0 1 1 1	27h

Un número con la cantidad de dígitos impar, justificaría el octeto más significativo a la derecha, por ejemplo 753 sería:

0 0 0 0 0 1 1 1	07h
0 1 0 1 0 0 1 1	53h

Como se puede ver los números decimales coinciden con los dígitos hexadecimales. Un octeto con valores en BCD nunca puede tener números hexadecimales entre A y F.

Para operar con octetos en este código se utilizan las instrucciones normales de suma y resta para registros de 8 bits y después de cada suma o resta es necesario ajustar el resultado a BCD. Por ejemplo al sumar 27 más 48 ocurriría lo siguiente:

```
27 -> BCD 00100111  27h
48 -> BCD 01001000  48h
```

la suma con ADD de estos dos octetos sería:

```
 00100111
+01001000
 01101111
```

resultado:

0 1 1 0 1 1 1 1	6Fh

si a este resultado se le suma 06h, quedaría:

```
 01101111
+00000110
 01110101
```

resultado:

0 1 1 1 0 1 0 1	75h

Que vuelve a estar en código BCD; efectivamente 27+48=75. Al sumar 06h al resultado, lo hemos transformado, de nuevo, en BCD.

Con una resta ocurriría lo mismo: por ejemplo 31 menos 15:

```
31 -> BCD 00110001  31h
15 -> BCD 00010101  15h
```

al restar con la instrucción SUB (recuerde que es una suma con el sustraendo complementado a 2) sería:

```
 00110001
+11101011
 00011100
```

resultado:

0 0 0 1 1 1 0 0	1Ch

si a este resultado se le suma FAh, sería:

```
 00011100
+11111010
 00010110
```

resultando:

0 0 0 1 0 1 1 0	16h

como se ve, vuelve a estar en BCD pues efectivamente 31–15=16. Esta vez hemos tenido que sumar FAh para pasar a BCD.

Todo este procedimiento parece complicado, pero existe una instrucción, DAA (Decimal Adjust in "A"), que añade la cantidad adecuada en cada octeto. Esta instrucción, que se explica a continuación, es la encargada de ajustar a BCD el octeto después de cada operación; con lo cual la única responsabilidad del programador es tener los operandos en BCD y aplicar la instrucción DAA después de cada operación, con esto obtendrá los resultados en BCD.

De todas formas, es muy improbable que tenga que utilizar esta instrucción en alguno de sus programas ya que para operar en decimal es más fácil utilizar el calculador de la ROM (lo veremos más adelante). Esta instrucción está pensada para pequeños sistemas construidos alrededor del Z-80, por ejemplo, para aplicaciones de control en tiempo real, donde no se dispone de un sofisticado Sistema Operativo que gestione las entradas y salidas en decimal.

La instrucción DAA utiliza como información inicial los indicadores en condición C, H y N.

El indicador C informa que existe acarreo en el octeto. Siempre que se presente, además de condicionar la cantidad que tiene que añadir, lo mantiene activo para poder añadirlo al octeto siguiente.

otro caso
N; pone 0 – siempre
C; pone 1 – si hay acarreo desde el bit 15
pone 0 – en cualquier otro caso

CICLOS DE MEMORIA:

4

CICLOS DE RELOJ:

15

EJEMPLO:

ADD IX,SP

Contenido del registro "SP"

(SP):	0 0 0 1 0 0 0 0	10h
	0 1 1 0 1 0 0 1	69h

Contenido del registro índice "IX"

(IX):	0 1 1 0 0 1 1 0	66h
	0 1 1 1 0 1 0 1	75h

Instrucción

ADD IX,SP:	1 1 0 1 1 1 0 1	DDh
	0 0 1 1 1 0 0 1	39h

Contenido del registro índice "IX" después de la ejecución

(IX):	0 1 1 1 0 1 1 0	76h
	1 1 0 1 1 1 1 0	DEh

Indicadores de condición después de la ejecución

S	Z		H		P/V	N	C
x	x	x	0	x	x	0	0

ADD IY, rr

OBJETO:

Sumar al contenido del registro índice "IY", el contenido del par de registros representados por "rr". El resultado se deja en el registro índice "IY". Observe que decimos "rr" en lugar de "pp" porque "pp" incluye al registro "IX" y "rr" incluye al "IY", por tanto, es posible multiplicar por dos el contenido de cualquiera de los dos registros índices (ADD IX, IX o ADD IY, IY) pero no es posible sumarlos los dos (ADD IX,IY o ADD IY,IX).

La codificación de "rr" es la siguiente:

rr	reg.
00	BC
01	DE
10	IY
11	SP

CODIGO DE MAQUINA:

1 1 1 1 1 1 0 1	FDh
0 0 r r 1 0 0 1	

INDICADORES DE CONDICION A LOS QUE AFECTA:

H; pone 1 – si hay acarreo desde el bit 11
pone 0 – en cualquier otro caso
N; pone 0 – siempre
C; pone 1 – si hay acarreo desde el bit 15
pone 0 – en cualquier otro caso

CICLOS DE MEMORIA:

4

CICLOS DE RELOJ:

15

EJEMPLO:

ADD IY,IY

Contenido del registro índice "IY"

(IY):	0 0 0 0 0 0 0 0	00h
	0 0 0 0 0 0 1 0	02h

Instrucción

ADD IY,IY:	1 1 1 1 1 1 0 1	FDh
	0 0 1 0 1 0 0 1	29h

Contenido del registro índice "IY" después de la ejecución

(IY):	0 0 0 0 0 0 0 0	00h
	0 0 0 0 0 1 0 0	04h

Indicadores de condición después de la ejecución

S	Z		H		P/V	N	C
x	x	x	0	x	x	0	0

El resultado de este ejemplo ha sido el de multiplicar por dos el contenido del registro índice "IY" (lo hemos sumado consigo mismo).

La activación de los indicadores de condición en las instrucciones de suma y resta de 16 bits, se hace según las siguientes reglas:

S: Este indicador sólo se contempla en las instrucciones de sumar y restar con acarreo y se pone el mismo valor que tenga el bit 15 del par de registros "HL" después de la ejecución.

Z: Este indicador sólo se contempla en las instrucciones de sumar y restar con acarreo. Se activa, valor igual 1, si todos los bits del par de registros "HL" son cero después de la ejecución.

H: Este indicador actúa igual que para las instrucciones de sumar y restar de 8 bits, con la diferencia de que el acarreo o no acarreo se define en el bit 11 del registro doble donde se deja el resultado.

P/V: Este indicador ctúa igual que en las instrucciones de suma y resta de 8 bits, con la diferencia de que el rango −128, 0, +127 es −32768, 0, + 32767. Estos límites son los valores mínimos o máximos que se pueden representar en complemento a 2 en dos octetos.

N: Este indicador no tiene significado para este grupo de instrucciones y se pone a 0 o 1, según sea suma o resta.

C: Este indicador actúa igual que en las instrucciones de sumar y restar de 8 bits, con la diferencia de que el acarreo o no acarreo se define en el bit 15 del registro doble donde se deja el resultado.

Grupo de Incremento y Decremento para 16 bits

Las instrucciones INC y DEC de 16 bits son básicamente iguales a las INC y DEC de 8 bits, las únicas diferencias son: que trabajan sobre registros dobles de 16 bits y que no afectan a los indicadores de condición.



OBJETO:

Añade uno al contenido del registro doble representado por "ss". La codificación de "ss" es la siguiente:

ss	reg
00	BC
01	DE
10	HL
11	SP

CODIGO DE MAQUINA:

0 0 s s 0 0 1 1

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

1

CICLOS DE RELOJ:

6

EJEMPLO:

Código Fuente	Hexadecimal	Decimal
OR A	B7	183
OR B	B0	176
OR C	B1	177
OR D	B2	178
OR E	B3	179
OR H	B4	180
OR L	B5	181
OR n	F6,n	246,n
OR (HL)	B6	182
OR (IX+d)	DD,B6,d	221,182,d
OR (IY+d)	FD,B6,d	253,182,d

Fig. 6-14. Tabla de codificación para el operador «OR».

Código Fuente	Hexadecimal	Decimal
XOR A	AF	175
XOR B	A8	168
XOR C	A9	169
XOR D	AA	170
XOR E	AB	171
XOR H	AC	172
XOR L	AD	173
XOR n	EE,n	238,n
XOR (HL)	AE	174
XOR (IX+d)	DD,AE,d	221,174,d
XOR (IY+d)	FD,AE,d	253,174,d

Fig. 6-15. Tabla de codificación para el operador «XOR».

después de la ejecución

S	Z		H		P/V	N	C
1	0	x	1	x	0	1	0

CCF: "Complement Carry Flag"; en inglés "Complementar el indicador de acarreo"



OBJETO:

Invierte el valor del indicador de condición (C) en el registro "F"; es decir, pasa a valer "0" si antes valía "1" y viceversa.

CODIGO DE MAQUINA:

0 0 1 1 1 1 1 1	3Fh

INDICADORES DE CONDICION QUE AFECTA:

H; mantiene su valor anterior
N; pone 0 – siempre
C; pone 1 – si C era cero antes de la ejecución
pone 0 – en cualquier otro caso

CICLOS DE MEMORIA:

1

CICLOS DE RELOJ:

4

EJEMPLO:

CCF

Valor del registro "F"

S	Z		H		P/V	N	C
0	0	x	1	x	0	0	1

Instrucción

CCF:	0 0 1 1 1 1 1 1	3Fh

Valor del registro "F" después de la ejecución

S	Z		H		P/V	N	C
0	0	x	1	x	0	0	0

No resulta alterado el contenido del acumulador ni de ningún otro registro.

SCF: "Set Carry Flag"; en inglés: "Poner a "1" el indicador de acarreo".

SCF

OBJETO:

Pone a 1 el valor del indicador de condición (C) del registro "F" independientemente del valor que tuviera antes.

CODIGO DE MAQUINA:

0 0 1 1 0 1 1 1	37h

INDICADORES DE CONDICION QUE AFECTA:

H; pone 0 – siempre
N; pone 0 – siempre
C; pone 1 – siempre

CICLOS DE MEMORIA:

1

CICLOS DE RELOJ:

4

EJEMPLO:

SCF

Valor del registro "F"

S	Z		H		P/V	N	C
0	0	x	1	x	0	0	0

Instrucción

SCF:	0 0 1 1 0 1 1 1	37h

Valor del registro "F" después de la ejecución

S	Z		H		P/V	N	C
0	0	x	1	x	0	0	1

Si el indicador de acarreo fuera "1" antes de la ejecución, la instrucción, lógicamente, no lo modificaría.

Tal vez, el lector haya echado en falta una instrucción que ponga a "0" el indicador de acarreo. El caso es que esa instrucción no existe, simplemente, porque no es necesaria. La secuencia "SCF CCF" nos asegura que el indicador (C) acabe valiendo "0", pero ocupa dos bytes; hay una forma más fácil de poner el acarreo a "0", simplemente haga: "AND A". El contenido del acumulador no variará, pero el indicador de acarreo se pondrá, con toda certeza, a cero.

El micro procesador Z80, debido posiblemente a sus características de origen, tiene prevista una forma de operar con las instrucciones aritméticas en decimal.

Para ello emplea el código BCD (Binary Coded Decimal), código decimal expresado en binario. Los operandos, en este caso, tienen que estar definidos en dicho código.

El código BCD consiste en expresar cada dígito decimal en cuatro bits, de tal forma que en cada octeto entren dos dígitos.

Por ejemplo el número 19 se expresaría:

0 0 0 1 1 0 0 1	19h

00101111 2Fh

INDICADORES DE CONDICION QUE AFECTA:

H; pone 1 – siempre
N; pone 1 – siempre

CICLOS DE MEMORIA:

1

CICLOS DE RELOJ:

4

EJEMPLO:

CPL

Contenido del registro "A"

(A): 10011011 9Bh

Instrucción

CPL: 00101111 2Fh

Contenido del registro "A" después de la ejecución

01100100 64h

Indicadores de condición después de la ejecución

S	Z		H		P/V	N	C
x	x	x	1	x	x	1	x

NEG, NEGate, negar en inglés. Por negar un número se entiende complementarlo a 2; el complemento a 2 es el complemento a 1 +1 y se utiliza como su negativo, por tanto, esta instrucción lo que hace realmente es cambiar de signo el contenido del acumulador (ver capítulo de sistemas de numeración).

NEG

OBJETO:

Complementa a dos el contenido del registro acumulador, dejando en el mismo el resultado.

CODIGO DE MAQUINA:

11101101 EDh
01000100 44h

INDICADORES DE CONDICION QUE AFECTA:

S; pone 1 – si el resultado es negativo
pone 0 – en cualquier otro caso
Z; pone 1 – si el resultado es cero
pone 0 – en cualquier otro caso
H; pone 1 – si no hay acarreo desde el bit 3
pone 0 – en cualquier otro caso
N; pone 1 – siempre
C; pone 1 – si el acumulador era 00h antes de la operación
pone 0 – en cualquier otro caso
P/V; pone 1 – si el acumulador era 80 antes de la operación
pone 0 – en cualquier otro caso

Observe que la forma de activarse los indicadores de condición recuerda a las instrucciones de restar; esto es porque lo que realmente hace esta instrucción es restar a 0 el contenido del acumulador (0-A). Recuerde que las instrucciones de restar lo primero que hacen es complementar a dos el sustraendo y después sumarle el minuendo; como el minuendo es siempre cero, el resultado es siempre el sustraendo complementado a 2. Como se puede deducir, a pesar de lo dicho, el indicador H siempre será 1. Este funcionamiento nos indica que el microprocesador utiliza para "negar" la misma circuitería que para "restar"; esto ocurre con muchas otras instrucciones y es muy lógico dado que, en un microprocesador, lo que se pretende es meter el mayor número de funciones en el menor espacio posible.

CICLOS DE MEMORIA:

2

CICLOS DE RELOJ:

8

EJEMPLO:

NEG

Contenido del registro "A"

(A): 01100101 65h

Instrucción

NEG: 11101101 EDh
01000100 44h

Operación:

```
                   01100101
complemento a 1 =  10011010
                 + 00000001
complemento a 2 =  10011011

                   00000000
                 + 10011011
      resultado =  10011011
```

Contenido del registro "A" después de la ejecución

10011011 9Bh

Indicadores de condición

INC HL

Contenido del par de registros "HL"

(H): 01100100 64h
(L): 11111111 FFh

Instrucción

INC HL: 00100011 23h

Contenido del par de registros "HL" después de la ejecución

(H): 01100101 65h
(L): 00000000 00h

El octeto alto de "HL" se ha incrementado en 1, dado que al incrementar el octeto bajo, éste ha pasado a valer "0". Vemos, por tanto, que el acarreo se transmite de forma automática desde el octeto bajo al alto.

INC IX

OBJETO:

Añade uno al contenido de registro indice "IX".

CODIGO DE MAQUINA:

11011101 DDh
00100011 23h

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

2

CICLOS DE RELOJ:

10

EJEMPLO:

INC IX

Contenido del registro indice "IX"

(IX): 11111111 FFh
11111111 FFh

Instrucción

INC IX: 11011101 DDh
00100011 23h

Contenido del registro indice "IX" después de la ejecución

(IX): 00000000 00h
00000000 00h

En este caso, el registro "IX" valía "FFFFh" antes de la instrucción, lo que hace que al sumarle "1", pase a valer "0"; no hay indicador de acarreo que nos indique esta circunstancia, pero ya veremos que no es necesario, ya que nunca tendremos que iterar un bucle o mover un puntero más de 65536 veces.

INC IY

OBJETO:

Añade uno al contenido del registro indice "IY".

CODIGO DE MAQUINA:

11111101 FDh
00100011 23h

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

2

CICLOS DE RELOJ:

10

EJEMPLO:

INC IY

Contenido del registro indice "IY"

(IY): 00000000 00h
00000000 00h

Instrucción

INC IY: 11111101 FDh
00100011 23h

Contenido del registro indice "IY" después de la ejecución

(IY): 00000000 00h
00000001 01h

Las instrucciones de decrementar funcionan igual que las anteriores, salvo que restan "1" en vez de sumarlo. Tampoco afectan a los indicadores.



OBJETO:

Decrementa uno al contenido del registro doble representado por "ss". La codificación de "ss" es la siguiente:

ss	reg
00	BC
01	DE
10	HL
11	SP

CODIGO DE MAQUINA:

0 0 s s 1 0 1 1

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

1

CICLOS DE RELOJ:

6

EJEMPLO:

```
DEC DE
```

Contenido del par de registros "DE"

(D):	0 1 1 0 0 1 1 0	66h
(E):	0 0 0 0 0 0 0 0	00h

Instrucción

DEC DE:	0 0 0 1 1 0 1 1	1Bh

Contenido del par de registros "DE" después de la ejecución

(D):	0 1 1 0 0 1 0 1	65h
(E):	1 1 1 1 1 1 1 1	FFh

Vemos, de nuevo, que el acarreo se ha vuelto a transmitir al octeto alto desde el bajo. Este valía cero, y el decrementarlo ha hecho que pasase a valer "FFh", por lo que el alto ha sido decrementado también.

DEC IX

OBJETO:

Decrementa uno al contenido del registro indice "IX".

CODIGO DE MAQUINA:

1 1 0 1 1 1 0 1	DDh
0 0 1 0 1 0 1 1	2Bh

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

2

CICLOS DE RELOJ:

10

EJEMPLO:

```
DEC IX
```

Contenido del registro indice "IX"

(IX):	0 0 0 0 0 0 0 0	00h
	0 0 0 0 0 0 0 0	00h

Instrucción

DEC IX:	1 1 0 1 1 1 0 1	DDh
	0 0 1 0 1 0 1 1	2Bh

Contenido del registro indice "IX" después de la ejecución

(IX):	1 1 1 1 1 1 1 1	FFh
	1 1 1 1 1 1 1 1	FFh

Vemos que el registro "IX" ha pasado de valer "0" a valer "FFFFh"; de nuevo, no hay indicador que ponga de manifiesto esta circunstancia pero, como dijimos antes, no es necesario.



DEC IY

OBJETO:

Decrementa uno al contenido del registro indice "IY".

CODIGO DE MAQUINA:

1 1 1 1 1 1 0 1	FDh
0 0 1 0 1 0 1 1	2Bh

INDICADORES DE CONDICION QUE AFECTA:

Ninguno

CICLOS DE MEMORIA:

2

CICLOS DE RELOJ:

10

EJEMPLO:

```
DEC IY
```

Contenido del registro indice "IY"

(IY):	0 0 0 0 0 0 0 0	00h
	0 0 0 0 0 0 0 1	01h

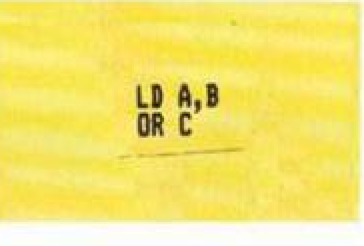
Instrucción

DEC IY:	1 1 1 1 1 1 0 1	FDh
	0 0 1 0 1 0 1 1	2Bh

Contenido del registro indice "IY" después de la ejecución

(IY):	0 0 0 0 0 0 0 0	00h
	0 0 0 0 0 0 0 0	00h

Esta vez, el registro "IY" vale cero después de la ejecución y no se activa el indicador de "cero" (Z). Si estuviéramos haciendo un bulce de más de 256 iteraciones, y utilizáramos un determinado registro como contador, sería útil poder comprobar cuándo este registro se hace "cero" para salir del bucle; la instrucción DEC en registros de 16 bits no afecta al indicador de "cero", pero es posible hacer un pequeño truco: supongamos que estamos usando el registro "BC" como contador de nuestro bucle, podemos saber si hemos llegado a "cero" mediante la siguiente operación:



```
LD A,B
OR C
```

Es decir, cargamos en "A" el contenido de "B" y le hacemos un "OR" con "C". Si el contenido de "BC" era "cero", el resultado de esta operación será también "cero", y se pondrá a "1" el indicador (Z) del registro "F".

Los usos más importantes de las instrucciones INC y DEC para registros de 16 bits son recorrer una tabla o zona de memoria desde un comienzo a un final o viceversa y establecer bucles con más de 256 iteraciones (los bucles se verán detenidamente en el siguiente capitulo).

Normalmente el rastreo de zonas de memoria se hace con registros índice o con el par de registros "HL". Un ejemplo muy sencillo sería mover una tabla de x octetos desde la posición de memoria MEMO1 a la posición de memoria MEMO2; la rutina que se puede hacer con estas instrucciones sería como indica la FIGURA 6-16.

COMIENZO → CARGAR IX CON MEMO 1 → CARGAR IY CON MEMO 2 → CARGAR B CON X-1 → CARGAR EN A OCTETO DE (IX+0) → CARGAR EN (IY+0) OCTETO DE A → INC IX → INC IY → DEC B → B=0 (NO: vuelve a CARGAR EN A OCTETO DE (IX+0); SI) → FIN

Fig. 6-16.

En este caso se trata sólo de desplazar los octetos de una zona a otra, pero muchas veces utilizaremos bucles similares para realizar determinada operación en toda una zona de memoria. Cuando se trate sólo de desplazar octetos, el microprocesador Z-80 posee unas instrucciones especiales que lo hacen de forma automática y mucho más deprisa; las estudiaremos con profundidad en un capítulo posterior.

Grupo de instrucciones aritméticas de uso general

En este grupo se encuadran unas instrucciones especiales que complementan a las anteriores, y que iremos viendo una a una, ya que guardan poca relación entre sí.

CPL, ComPLement, complemento en inglés. Por complementar a 1 se entiende invertir el valor de un número binario, más sencillo, cambiar todos los unos por ceros y los ceros por unos.

CPL

OBJETO:

Complementa a uno el contenido del registro acumulador, esto es, invierte el valor de todos sus bits.

CODIGO DE MAQUINA:



DISPONIBLE PARA ZX SPECTRUM
Sound-on-Sound
Sound on Sound es una marca registrada producida y distribuida por Iberofón, s. a.
Telf. 671.22.00 / 04 / 08 / 12 / 16
JUEGA CON EL FUTURO
SOFTWARE
THE ROCKY HORROR SHOW
SPECTRUM 48K
DON'T DREAM IT... PLAY IT!
CASINO ROYAL
SPECTRUM 48K
Sound-on-Sound
¡¡¡NO LO SUEÑES!!! ¡JUEGALO!
SIENTE LA EMOCION DE LO DESCONOCIDO
CORRE TU PROPIO RIESGO
SALVA A TU COMPAÑERO/A ATRAPADO/A
REUNE LOS FRAGMENTOS DEL CUADRO
SON TU AMULETO
"BLACK JACK" ¡¡¡EL JUEGO DE CARTAS MAS EMOCIONANTE!!!
¡¡¡CASINO ROYAL!!!
"LA EMOCION DEL JUEGO POR TODO LO ALTO"
¡¡¡POR FIN EN CASTELLANO!!!
LA PRIMERA COMEDIA MUSICAL EN VIDEO-JUEGO
Graphis, S.A. Publicidad

GENERADOR DE ROTULOS (I)

Miguel SEPULVEDA y Adolfo PEREZ

En este artículo vamos a explicar un método para generar pantallas de rótulos que, entre otras utilizaciones sirve para grabar presentaciones de cintas o de reportajes para vídeo.

El programa consta fundamentalmente de dos partes. En una de ellas, el trabajo consistirá en diseñar los tipos de pantallas que queramos, y en la otra, la forma en que queremos que estas pantallas nos aparezcan después.

Una vez que nosotros hayamos diseñado una cabecera o la presentación de un reportaje, lo guardaremos en memoria y luego lo presentaremos de una de las cinco formas posibles que nos permite el programa que vamos a desarrollar.

Las pantallas podremos generarlas con dos tipos distintos de letras, bien con letras normales (pero de tamaño variable), o bien con letras en tres dimensiones (tamaño fijo). Cualquier pantalla podremos generarla con uno de estos tipos o con una mezcla de ambos. Las pantallas generadas podremos presentarlas en la TV durante un tiempo determinado por nosotros, y podremos presentar tantas como queramos, sin más limitación que el tamaño de la memoria. No obstante, como para almacenar utilizaremos rutinas que comprimen los archivos de imagen y de atributos, el número de pantallas que podremos guardar será muy elevado.

Un ejemplo de una pantalla presentada por el monitor de TV podría ser la de la figura 1. En ella observamos que hay letras en tres dimensiones y letras normales de distinto tamaño y distinta separación entre ellas. Este sería un ejemplo de una pantalla. El programa podrá ser capaz de presentar este rótulo durante el tiempo que nosotros deseemos y en la forma que queramos, borrarlo y seguir presentando otros sucesivamente tal como sucede en la presentación de una película.

Además de los rótulos como el de la figura 1, también podremos generar textos, que irán apareciendo por la parte inferior de la pantalla y subiendo scan a scan hasta desaparecer. El tamaño del texto, también vendrá impuesto por la cantidad de memoria que nos quede libre.

Nosotros hemos desarrollado el programa pensando en su utilidad para el vídeo, pero sería igualmente útil para cualquier otra aplicación que se le quisiera dar como podría ser por ejemplo, la presentación de cabeceras de programas, escaparates de comercios, etc.

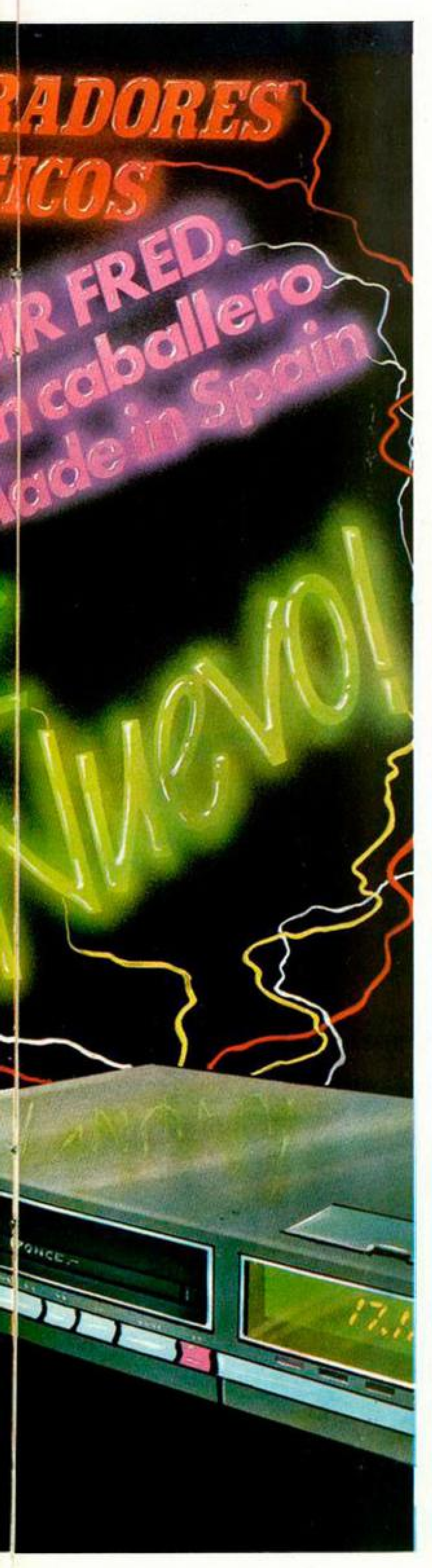
Antes de pasar a explicar el programa, un par de consideraciones. Sabemos que en memoria, los archivos de imagen y de atributos ocupan zonas consecutivas, el primero empieza en la posición 4000h, y el segundo en la 5800h, ocupando 1800h el de imagen, y 300h el de atributos. Sabemos también, que podemos considerar la pantalla del TV bajo dos puntos de vista: baja y alta resolución. En baja resolución la pantalla estaría compuesta por 24 filas y 32 columnas. En alta resolución podemos dividir cada fila y cada columna en ocho, de forma que la pantalla estaría compuesta por 8*24=192 filas y 8*32=256 columnas. Dentro de este conjunto de filas y columnas, vamos a considerar como punto (0,0) el ángulo superior izquierdo, por tanto, la fila 1 se encontrará en la parte superior y la primera columna en la parte izquierda de la pantalla. Cada fila en alta resolución la llamaremos scan.

Teniendo en cuenta lo anterior, vamos a pasar a explicar el programa BASIC que nos permite generar y presentar las pantallas.

Una vez cargado y ejecutado el programa 1, nos aparece en pantalla el menú principal con 7 opciones:

1. Crear pantallas de rótulos.
2. Introducir textos.
3. Preparar salida.
4. Ejecución de lo preparado.
5. Limpia la memoria.
6. Información de lo creado.
7. Salir del programa.





HOLA
ESTO ES UN EJEMPLO
de composicion de
PaNtAllaS
CON
distintas LETRAS
QUE TAL ?

Fig. 1.

OPCIONES

1. CREAR PANTALLAS DE ROTULOS

Obviamente, ésta debe ser la primera opción que deberemos elegir, pues sin ella no tienen sentido las demás.

Esta opción se refiere a la creación y almacenamiento en memoria de un número determinado de pantallas. Estas pantallas podrán estar configuradas con distintos tipos de letras, tamaños y separaciones entre caracteres y líneas.

Las distintas separaciones entre caracteres se refiere a distintos scans, ya que dentro de un scan la separación será idéntica para todos los caracteres que lo compongan.

En el listado BASIC esta opción comprende desde la línea 5Ø hasta la línea 31Ø ambas inclusive. Lo primero que hace el programa es pedirnos el número de pantallas que vamos a crear (variable pt) y esto servirá como límite superior del bucle FOR-NEXT que va desde la línea 6Ø hasta la 22Ø. En cada pasada de este bucle, lo que se hace es la generación de una pantalla.

En primer lugar, en las líneas 7Ø y 8Ø, especificamos los valores de los atributos de pantalla. A continuación, en la línea 9Ø, nos pide el tipo de letras que van a componer este rótulo, si son letras en tres dimensiones o no.

En la línea 250, el programa nos pide el número de scan inicial a partir del cual empezará a dibujar el carácter.

Hemos dicho que en alta resolución podríamos considerar la pantalla como una matriz de 176 filas y 256 columnas. Consideraremos sólo 168 filas porque la línea 22 de pantalla (en baja resolución) no la utilizaremos. Cada fila podremos considerarla como si estuviese formada por 256 bits o puntos. Cada una de estas filas, ya dijimos anteriormente que constituyen un scan.

Consideraremos como inicio de scans el ángulo superior izquierdo, de modo que el scan Ø será la línea horizontal más alta de la pantalla.

Un carácter en tres dimensiones necesita un espacio de 30*30, es decir, 3Ø trozos de scan, cada uno de los cuales tiene un tamaño de 3Ø bits. Es fácil entonces deducir, que en cada línea horizontal tendríamos un máximo de 256/30=8,5 caracteres. Como los caracteres tienen una separación fija, estos 8,5 caracteres quedan reducidos a un máximo de 7. A su vez, como en vertical también ocupan 30 bits, resulta que nos cabrían 5 líneas. Por tanto, en cada pantalla podremos meter 5 líneas de 7 caracteres cada una. Nos referimos claro está, a caracteres completos.

Volviendo a la línea 25Ø, lo que se nos pide, es el scan a partir del cual se comenzará a dibujar el carácter. Teniendo en cuenta lo anteriormente dicho, el número más alto que podríamos dar para que el carácter quepa entero, sería el 13Ø. A continuación, debemos dar el primer rótulo que compone esa pantalla. Decimos el primer rótulo, puesto que una pantalla en 3D podría estar compuesta hasta por cinco rótulos, cada uno de ellos con un máximo de siete caracteres. Esto se comprueba en la línea 26Ø.

Después de teclear el rótulo, éste aparece en pantalla en el lugar elegido. Una vez presentado el rótulo, aparece en la línea 22 de pantalla (en baja resolución) la siguiente información:

M=Menú R=Reponer S=Salvar C=Cont

MENU DE OPCIONES

M Lo que hace es presentarnos el menú principal y empezar el proceso como si se tratase de uno nuevo. Por tanto, todos los rótulos que hayamos introducido desaparecerán, y no se almacenará nada en memoria.

R Empezaremos dando los atributos de una pantalla. Tampoco hemos almacenado nada, pero seguimos dentro del proceso en el que estamos, es decir, no tendríamos que empezar especificando el número de pantallas a crear.

C Lo que se hace es seguir con la pantalla que estábamos hasta completarla, es decir, volveríamos a repetir el proceso para otro rótulo, pudiendo cambiar incluso de tipo de letra.

S Aquí hace una llamada a una rutina en código máquina que tiene por finalidad comprimir la pantalla y almacenarla.

A continuación se incrementa la variable del bucle FOR-NEXT y se repite el proceso hasta completar el número de pantallas especificado.

Veamos un ejemplo. Supongamos que queremos grabar una cinta con el título «ASTERIX OBELIX Y LOS ROMANOS» utilizando letras en tres dimensiones. Si observamos la figura 3 que es como nos va a quedar la pantalla, vemos que dicha pantalla está compuesta por cuatro rótulos diferentes, por lo tanto, tendremos que componerla en cuatro pasadas distintas. La forma de operar sería la siguiente: en primer lugar elegiríamos la opción 1 del Menú principal. A la pregunta de cuántas pantallas queremos crear, responderíamos que una. A la siguiente pregunta, responderíamos afirmativamente (nos dice si queremos las letras en 3D). Seguidamente le damos el scan donde comenzaran las letras del primer rótulo. En el caso de nuestro ejemplo le dimos el valor cero (cualquier otro valor sería igualmente válido, sólo que empezaría más abajo). A continuación le damos el primer rótulo «ASTERIX». Después de teclearlo, el programa nos presenta el rótulo en el lugar elegido y además en la línea 22 de la pantalla de TV nos aparecerán las letras M, R, S y C. Hemos de teclear una de estas opciones. Nosotros hemos de dar C puesto que la pantalla no está completa y el proceso tiene que seguir. Lo mismo se repetirá para cada uno de los tres rótulos restantes, debiendo especificar en cada uno de ellos, el scan a partir del cual comenzarán. En nuestro caso, los rótulos empiezan en los scans Ø, 42, 84 y 126. De esta manera nos salen más o menos centrados los rótulos. Con el número del scan, posicionamos los rótulos en líneas horizontales. Para situarlos en líneas verticales no hemos de preocuparnos, puesto que dependiendo del número de caracteres que compongan el rótulo, el mismo programa se encarga de iniciarlo de forma que el rótulo quede centrado. Cuando el cuarto rótulo haya sido tecleado, ya no hemos de dar C puesto que la pantalla está completa, sino S para que todos los datos que constituyen la pantalla sean almacenados en memoria después de comprimidos. Después de teclear S, se nos volverá a presentar el menú principal y elegiremos otra opción cualquiera.

Si durante la creación de la pantalla anterior pulsamos M o R, todo lo introducido hasta ese momento se borra. Si pulsamos S, se almacena todo lo hecho hasta ese momento dando por completada la pantalla.

COMO CONECTAR EL SPECTRUM AL VIDEOGRABADOR

Para todos los usuarios de vídeo que quieran hacer uso de este programa, una posible conexión entre los aparatos podría ser: sintonizamos un canal del aparato de vídeo al canal por el que sale el Spectrum en la TV. Una vez ejecutado el programa y teniendo las pantallas y textos compuestos, grabaremos desde el vídeo por el canal elegido.

PROGRAMA 1

```
  1 CLEAR 31999: REM A. Perez y M. Sepulveda
  2 LOAD ""CODE 63392,1053
 10 DIM m$(19,2): DIM n(19,2): DIM k(19,2): DIM c$(19,2): LET x=1: LET p$="OR TP D.COM D.FIN N. BY A TM C TC": GO SUB 12: GO TO 20
 12 FOR n=1 TO 19: LET m$(n)="  ": LET c$(n)="  ": LET n(n,1)=0: LET n(n,2)=0: LET k(n,1)=0: LET k(n,2)=0: NEXT n
 13 LET x=1: LET d=32000: RETURN
 15 PRINT AT 0,0;p$
 16 FOR n=1 TO x-1: PRINT AT n,0;n;TAB 3;m$(n);TAB 6;n(n,1);TAB 12;n(n,2);TAB 18;n(n,2)-n(n,1);TAB 23;c$(n,1);TAB 25;k(n,1);TAB 28;c$(n,2);TAB 30;k(n,2): NEXT n
 17 RETURN
 20 PAPER 7: INK 0: CLS : PRINT AT 2,9;"MENU PRINCIPAL"''''"1- Crear pantallas de rotulos"''"2- Introducir texto"''"3- Preparar salida por pantalla"''"4- Ejecucion de lo preparado"''"5- Limpiar memoria"''"6- Informacion de lo creado"''"7- Salir del programa"
 30 PAUSE 0: LET a$=INKEY$: CLS
 40 IF a$<>"1" THEN GO TO 400
 50 INPUT AT 0,0;"Cuantas pantallas quiere ";pt
 60 FOR n=1 TO pt
 70 INPUT AT 0,0;"Color del PAPER ";p;AT 0,10;"BORDER ";b
 80 BORDER b: PAPER p: CLS
 90 INPUT AT 0,0;"Lo quiere en 3-D ";LINE a$: IF a$="s" THEN GO TO 250
100 INPUT AT 0,0;"Num. del scan (0 a 168) ";ns: INPUT AT 0,0;"Altura del caracter ";al;AT 0,0;"Anchura del caracter ";an;AT 0,0;"Separacion entre caracteres ";sp
110 INPUT AT 0,0;"Rotulo: "; LINE b$
120 INPUT AT 0,0;"INK= ";i;" BRIGHT= ";b;" FLASH= ";f: LET at=i+p*8+b*64+f*128
130 POKE 65280,al: POKE 65281,an: POKE 65282,sp: POKE 65283,INT ((256-(sp*(LEN b$-1)+an*8))/2): POKE 65284,ns: POKE 65285,at: POKE 65286,LEN b$: POKE 65287,INT ((sp*LEN b$)/8+.9)
135 FOR z=1 TO LEN b$: POKE 65293+z,CODE b$(z): NEXT z
140 RANDOMIZE USR 64195
150 PRINT #1;AT 0,0;"M=Menu R=Reponer S=Salvar C=Cont"
160 PAUSE 0: LET a$=INKEY$
170 IF a$="m" THEN GO TO 20
180 IF a$="r" THEN GO TO 70
190 IF a$="c" THEN GO TO 90
200 IF a$<>"s" THEN GO TO 160
210 PRINT #1;AT 0,0;"                              ": LET n(x,1)=d: RANDOMIZE d: LET d=USR 63630: LET n(x,2)=d: LET m$(x)="RO": LET x=x+1
220 NEXT n: GO TO 20
250 INPUT AT 0,0;"Num. del scan (0 a 130) ";ns: IF ns>130 THEN GO TO 250
260 INPUT AT 0,0;"Rotulo en 3D (max. 7) "; LINE b$: LET t=LEN b$: IF t<1 OR t>7 THEN GO TO 260
265 LET m=INT ((256-t*32)/2-4)
270 PRINT INK p;AT 21,0;b$
280 FOR z=0 TO t*8-1: FOR y=0 TO 7
290 IF POINT (z,y) THEN PLOT z*4+m,y*4+138-ns: DRAW 4,0: DRAW 0,4: DRAW -4,0: DRAW 0,-3: DRAW 3,0: DRAW 0,2: DRAW -2,0: DRAW 0,-1: DRAW 2,0: DRAW -2,-2: DRAW 5,5: DRAW 0,4: DRAW 0,-4: DRAW 4,0: DRAW 0,4: DRAW 0,-4: DRAW -5,-5: DRAW 0,4: DRAW 5,5: DRAW -4,0: DRAW -5,-5
300 NEXT y: NEXT z
305 PRINT AT 21,0;"          "
310 GO TO 150
400 IF a$<>"2" THEN GO TO 500
405 LET n(x,1)=d: INPUT AT 0,0;"BORDER= ";b;" PAPER= ";p;" INK= ";i: POKE d,b: POKE d+1,p: POKE d+2,i: LET d=d+3: BORDER b: PAPER p: INK i: CLS
410 INPUT AT 0,0;"Margen izquierdo ";mi;AT 0,7;" derecho ";md: LET md=31-md
420 POKE 23692,255: PRINT AT 21,0;"                              ": LET li=mi: PRINT #1;AT 0,0;"DEL=Borra caracter EDIT=Finaliza"
425 FOR z=0 TO mi-1: POKE d,32: LET d=d+1: NEXT z
430 PRINT AT 21,li;"█": PAUSE 0: LET a$=INKEY$
440 IF a$=CHR$ 12 AND li>mi THEN LET d=d-1: PRINT AT 21,li;" ": LET li=li-1: GO TO 430
450 IF a$=CHR$ 7 THEN POKE d,13: POKE d+1,13: POKE d+2,255: LET d=d+3: LET n(x,2)=d: LET m$(x)="TE": LET x=x+1: GO TO 20
460 IF CODE a$=13 THEN POKE d,13: LET d=d+1: PRINT AT 21,li;" ": PRINT : GO TO 420
470 PRINT AT 21,li;a$: POKE d,CODE a$: LET d=d+1: LET li=li+1: IF li>md THEN PRINT : POKE d,13: LET d=d+1: GO TO 420
480 GO TO 430
500 IF a$<>"3" THEN GO TO 650
510 GO SUB 15
530 PRINT AT 21,0;"V=Ver. H=Hor. C=Cir. O=Lh. E=Lv."
550 INPUT AT 0,0;"Numero de orden= ";k: IF k>=x THEN PRINT AT 21,0;"[illegible]": PAUSE 0: GO TO 20
555 IF m$(k)="TE" THEN LET d$="t": LET c$(k,1)=d$: GO TO 590
560 INPUT AT 0,0;"Codigo de abertura= "; LINE d$: IF d$<>"v" AND d$<>"h" AND d$<>"c" AND d$<>"o" AND d$<>"e" THEN GO TO 560
570 LET c$(k,1)=d$: INPUT AT 0,0;"Tiempo de presentacion= ";t: IF t>99 THEN GO TO 570
580 LET k(k,1)=t: INPUT AT 0,0;"Codigo de cierre= "; LINE d$: IF d$<>"v" AND d$<>"h" AND d$<>"c" AND d$<>"o" AND d$<>"e" AND d$<>"d" THEN GO TO 580
590 LET c$(k,2)=d$: INPUT AT 0,0;"Tiempo de cierre= ";t: IF t>99 THEN GO TO 590
600 LET k(k,2)=t: PRINT AT k,23;c$(k,1);TAB 25;k(k,1);TAB 28;c$(k,2);TAB 30;k(k,2): GO TO 550
650 IF a$<>"4" THEN GO TO 750
655 PAPER 0: BORDER 0: INK 0: CLS
660 FOR n=1 TO x-1
665 IF c$(n,1)=" " THEN GO TO 730
670 LET z=n(n,1): IF c$(n,1)<>"t" THEN GO TO 680
673 BORDER PEEK z: PAPER PEEK (z+1): INK PEEK (z+2): CLS : LET z=z+3
676 RANDOMIZE z: LET z=USR 64027: GO TO 719
680 RANDOMIZE z: LET z=USR 63970
690 IF c$(n,1)="v" THEN RANDOMIZE USR 63599: GO TO 700
692 IF c$(n,1)="h" THEN RANDOMIZE USR 63442: GO TO 700
694 IF c$(n,1)="c" THEN RANDOMIZE USR 63754: GO TO 700
696 IF c$(n,1)="o" THEN RANDOMIZE USR 63626: GO TO 700
698 RANDOMIZE USR 63487
700 PAUSE k(n,1)*50
710 IF c$(n,2)="v" THEN RANDOMIZE USR 63538: GO TO 720
712 IF c$(n,2)="h" THEN RANDOMIZE USR 63392: GO TO 720
714 IF c$(n,2)="c" THEN RANDOMIZE USR 63704: GO TO 720
716 IF c$(n,2)="o" THEN RANDOMIZE USR 63552: GO TO 720
718 IF c$(n,2)="e" THEN RANDOMIZE USR 63416: GO TO 720
719 FOR z=1 TO 176: RANDOMIZE USR 64121: NEXT z: PAPER 0: CLS
720 PAUSE k(n,2)*50
730 NEXT n: GO TO 20
750 IF a$="5" THEN GO SUB 12: GO TO 20
760 IF a$<>"6" THEN GO TO 800
770 GO SUB 15
790 PRINT AT 21,0;"[illegible]": PAUSE 0: GO TO 20
800 IF a$<>"7" THEN GO TO 20
810 STOP
```

Profesor particular

POLINOMIOS III

Arturo LOBO y J. J. LEON

El programa de esta semana, tercero de la serie de polinomios, está especialmente indicado para los estudiantes de 3.º de B.U.P. y permite representar paso a paso polinomios o cocientes de polinomios tal y como vosotros mismos lo haríais.: hallando derivadas, ceros, máximos y mínimos, asíntotas, paridad o imparidad, etc., y construyendo un cuadro de valores de la función y sus derivadas en los intervalos apropiados. Por limitaciones de espacio no explicamos teóricamente algunos de los cálculos que va haciendo pero proporciona todos los resultados intermedios necesarios y te conduce ordenadamente a la solución final.

En resumen, si quieres aprender de verdad a representar funciones, nada mejor que, provisto del libro de matemáticas, acompañar al ordenador en la resolución del problema. Descubrirás que, en menos tiempo del que crees, todas tus posibles dudas se habrán disipado. Ah, y recuerda: este programa sólo funciona acompañado de los dos ya publicados de esta misma serie.

TODAS LAS LETRAS MAYUSCULAS SUBRAYADAS DEBERAN TECLEARSE EN MODO GRAFICO

```
 105 LET fal=0: FOR i=0 TO 23: READ a: POKE USR "j"+i,a: NEXT i
 115 DATA 0,0,16,16,124,16,124,0,0,0,102,153,153,153,102,0,0,0,8,4,126,4,8,0
3279 IF fal=1 THEN IF jee=1 THEN LET j$=STR$ d(1,1,1): RETURN
3305 IF fal=1 THEN RETURN
3940 IF CODE a$=13 THEN GO TO 5000
4767 IF fla=1 THEN RETURN
5000 CLS : INK 2: PLOT 0,121: DRAW 0,-35: DRAW 255,0: DRAW 0,35: DRAW -255,0: INK 1
5010 PRINT AT 5,2;"Opcion ";AT 8,0;"11)REPRESENTACION DE POLINOMIOS     Y COCIENTES DE POLINOMIOS"
5020 PRINT AT 17,1;"(Si quieres la opcion pulsa s)": PAUSE 0
5030 IF INKEY$<>"s" AND INKEY$<>"S" THEN GO TO 190
5032 IF fal=2 THEN GO TO 5600
5035 DIM g(21,3,2): LET fla=1
5040 INPUT " Polinomio o cociente? (p/c) "; LINE a$: IF a$<>"p" AND a$<>"c" THEN GO TO 5040
5050 LET coc=a$="c": IF NOT coc THEN GO SUB 4400
5113 IF NOT coc AND pol=1 THEN LET z$=p$: GO TO 5120
5115 IF NOT coc THEN LET z$=q$: GO TO 5120
5117 LET z$="("+p$+")/("+q$+")"
5120 REM
5125 INK 5: CLS : PRINT #0;"-La funcion es  f(x)=";z$;"-Ahora voy a calcular las deri- vadas (te aviso cuando termine)"
5130 PRINT #0;AT 7,11; FLASH 1;"CALCULANDO"
5140 IF coc THEN GO TO 5200
5160 GO SUB 4000: FOR i=1 TO 21: FOR j=1 TO 3: LET g(i,j,1)=d(i,j,1): NEXT j: NEXT i
5170 LET col=7: LET der=1: GO SUB 4240: GO SUB 1200: LET t$=b$: LET der=2: GO SUB 4240: GO SUB 1200: LET u$=b$: GO TO 5500
5200 LET fla=1: LET ope=5: LET nd5=1
5205 FOR i=1 TO 21: FOR j=1 TO 2: LET d(i,1,j)=p(i,j+2): LET g(i,1,j)=d(i,1,j): NEXT j: NEXT i
5210 GO SUB 4700: LET der=2: LET gru=5: GO SUB 4100: LET col1=7: LET o$=""
5220 FOR i=21 TO gra3 STEP -1: IF ABS p(i,7)>1e-4 THEN GO TO 5240
5230 NEXT i
5240 LET col2=3+(i<=gra3): GO SUB 4260
5250 LET col=3+(i<gra3): GO SUB 4240: GO SUB 1600: LET col=4: LET der=0: GO SUB 4220: GO SUB 4100: IF gra3<=1 THEN GO TO 5300
5260 FOR i=1 TO gra3: LET d(i,1,5)=d(i,1,5)/d(gra3,1,5): NEXT i
5270 GO SUB 4240: LET col1=col2-2: LET col2=3: GO SUB 4260: GO SUB 1700: FOR i=1 TO 21: LET g(i,nd5+1,1)=p(i,7): NEXT i
5280 LET der=2: LET col=3: GO SUB 4240: GO SUB 1700: FOR i=1 TO 21: LET g(i,nd5+1,2)=p(i,7): NEXT i: GO TO 5400
5300 FOR i=1 TO 21: LET g(i,nd5+1,1)=p(i,col-2): LET g(i,nd5+1,2)=p(i,5-col2): NEXT i
5400 IF nd5=1 THEN FOR i=1 TO 21: LET p(i,3)=g(i,2,1): LET p(i,4)=g(i,2,2): NEXT i: LET nd5=2: GO TO 5210
5405 LET mam=3
5410 FOR i=1 TO 21: LET p(i,7)=g(i,mam,1): NEXT i: GO SUB 1200
5420 LET t$="("+b$+")/("
5430 FOR i=1 TO 21: LET p(i,7)=g(i,mam,2): NEXT i: GO SUB 1200
5440 LET t$=t$+b$+")"
5450 IF mam=3 THEN LET mam=2: LET u$=t$: GO TO 5410
5500 INK 1: CLS : BEEP .3,10: BEEP .3,5: BEEP .3,14: BEEP .3,10
5510 LET fla=0: PRINT "-Estos son los resultados:";"  f";d$;"=";z$;"  f'";d$;"=";t$;"  f''";d$;"=";u$
5514 IF coc THEN LET pol=1
5515 FOR i=1 TO 21: FOR j=1 TO coc+1: LET p(i,1+pol(+j)=g(i,1,j): NEXT j: NEXT i
5520 PRINT "-Ahora debes ir a las opciones 9y 10 y hallar los ceros de f, f' y f''"
5525 IF coc THEN PRINT "-Halla tambien los ceros del de-nominador de f"
5530 PRINT "  *Copia antes los resultados": IF coc THEN PRINT "-Recuerda que los ceros de un cociente son los del numerador"
5540 PRINT : PRINT "  *Cuando los tengas vuelve": GO SUB 4480: LET fal=2: GO TO 200
5600 CLS : INPUT "Ya tienes los ceros? (s/n) "; LINE a$: IF a$<>"s" AND a$<>"S" THEN GO TO 200
5605 LET fal=1: LET z$="": DIM a(25): DIM b(25): DIM c(25): DIM e(25)
5610 LET z4=0: INPUT "Cuantos ceros tiene f? ";z1: IF z1>0 THEN FOR i=2 TO z1+1: INPUT "Introduce un cero ";a(i): NEXT i
5620 INPUT "Cuantos ceros tiene f'? ";z2: IF z2>0 THEN FOR i=1 TO z2: INPUT "Introduce un cero ";b(i+z1+1): NEXT i
5630 INPUT "Cuantos ceros tiene f''? ";z3: IF z3>0 THEN FOR i=1 TO z3: INPUT "Introduce un cero ";c(i+1+z1+z2): NEXT i
5635 IF coc THEN INPUT "Cuantos ceros tiene el denomi- nador de f? ";z4: IF z4>0 THEN FOR i=1 TO z4: INPUT "Introduce un cero ";e(i+1+z1+z2+z3): NEXT i
5638 LET der=0: LET gru=1: FOR j=1 TO coc+1: FOR k=1 TO 3: FOR i=1 TO 21: LET d(i,1,1)=g(i,k,j)
5640 NEXT i: GO SUB 4100: LET je=gra3+(gra3=0): GO SUB 3279: GO SUB 9999: NEXT k: NEXT j
5642 CLS : GO SUB 6650: GO SUB 4400: CLS : GO SUB 6500: GO SUB 4400
5695 LET der=0: LET gru=1
5700 PRINT #0;"Vamos a hacer un cuadro...(pulsa)": PAUSE 0: CLS
5720 PRINT "Intervalo    valor  valor  valor" "o punto      de f   de f'  de f''"
5730 PLOT 0,156: DRAW 255,0: PLOT 92,0: DRAW 0,175: PLOT 148,0: DRAW 0,175: PLOT 204,0: DRAW 0,175
5740 LET zz=z1+z2+z3+z4: IF zz=0 THEN GO TO 5800
5750 DIM f(zz): FOR i=2 TO zz+1: LET f(i-1)=a(i)+b(i)+c(i)+e(i): NEXT i
5760 FOR k=1 TO zz-1: FOR i=1 TO zz-k: IF f(i)>f(i+1) THEN LET j=f(i): LET f(i)=f(i+1): LET f(i+1)=j
5770 NEXT i: NEXT k
5773 FOR i=1 TO zz-1: IF f(i)=f(i+1) THEN FOR j=i+1 TO zz-1: LET f(j)=f(j+1): NEXT j: LET zz=zz-1: GO TO 5773
5775 NEXT i
5780 IF NOT coc THEN LET l$="1": LET m$=l$: LET n$=l$
5800 PRINT AT 3,0;"-";CHR$ 154;TAB 5;CHR$ 155;: LET eai=f(1)-1
5810 DEF FN a(x)=INT ((VAL f$/VAL l$)*100+.5)/100
5820 DEF FN b(x)=INT ((VAL g$/VAL m$)*100+.5)/100
5830 DEF FN c(x)=INT ((VAL h$/VAL n$)*100+.5)/100
5840 FOR i=1 TO zz: LET x=(f(i)+eai*PI)/(1+PI): PRINT f(i);TAB 14;CHR$ (43+2*(FN a()<0));TAB 21;CHR$ (43+2*(FN b()<0));TAB 28;CHR$ (43+2*(FN c()<0)): LET x=f(i)
5850 PRINT TAB 2;x;TAB 12;: GO SUB 6950: PRINT x;TAB 5;CHR$ 155;: LET eai=f(i)
5860 NEXT i: PRINT "+";CHR$ 154;
5870 LET x=f(zz)+1: PRINT TAB 14;CHR$ (43+2*(FN a()<0));TAB 21;CHR$ (43+2*(FN b()<0));TAB 28;CHR$ (43+2*(FN c()<0))
5880 PRINT #0;"Comprueba este cuadro...": GO SUB 4400
5890 DEF FN x(x)=((256*x)/(3*max)+128)*(2*ABS x<(3*max))+255*(2*x>3*max)
5900 DEF FN y(x)=((256*FN z(x))/(3*max)+88)*(32*ABS FN z(x)<(33*max))+175*(32*FN z(x)>=33*max)
5910 LET max=4: FOR i=1 TO zz
5920 IF ABS f(i)>max THEN LET max=ABS f(i)
5925 NEXT i: FOR i=1 TO z2: IF ABS FN z(b(i+1+z1))>max*5/6 THEN LET max=ABS FN z(b(i+1+z1))*6/5
5930 NEXT i: IF ABS aa>max THEN LET max=ABS aa
5940 IF bb<>PI THEN IF ABS (cc/bb)>max THEN LET max=ABS (cc/bb)
5950 CLS : PRINT "Vamos a dibujar maximos, minimos,ceros y asintotas": PAUSE 100: CLS
5960 GO SUB 6800: IF z1 THEN FOR i=1 TO z1: CIRCLE FN x(a(i+1)),88,2: BEEP .1,15: NEXT i
5970 IF z2 THEN FOR i=1 TO z2: CIRCLE FN x(b(i+z1+1)),FN y(b(i+1+z1)),1: BEEP .1,9: NEXT i
5980 IF bb<>PI THEN GO SUB 6900: PLOT x1,y1: DRAW INK 2;x2-x1,y2-y1: BEEP .2,0
5990 IF aa<>PI THEN PLOT 0,256*aa/3/max+88: DRAW INK 2;255,0: BEEP .2,0
5995 IF z4 THEN FOR i=1 TO z4: PLOT FN x(e(i+1+z2-z4)),0: DRAW INK 2;0,175: BEEP .2,0
6000 GO SUB 4450: PRINT #0;AT 1,0;"Ahora dibujamos la funcion"
6010 GO SUB 6750: GO SUB 4480: GO TO 200
6500 CLS : PRINT "ASINTOTAS";AT 3,2;"-Horizontales": LET i=21
6510 IF NOT (g(i,1,1) OR g(i,1,2)) THEN LET i=i-1: GO TO 6510
6520 IF g(i,1,1)*g(i,1,2) THEN LET aa=g(i,1,1)/g(i,1,2): PRINT "y=";aa: GO TO 6540
6530 PRINT "NO HAY": LET aa=PI
6540 PRINT AT 5,2;"-Oblicuas"
6550 IF g(i,1,1)<>0 AND aa=PI AND ... THEN IF g(i-1,1,2)<>0 THEN GO TO 6570
6560 LET bb=PI: PRINT "NO HAY": GO TO 6590
6570 LET bb=g(i,1,1)/g(i-1,1,2): LET cc=0: LET dd=g(i-1,1,1)/g(i-1,1,2): IF i>2 THEN LET cc=dd-bb*g(i-2,1,2)
6580 PRINT bb;d$;: IF cc THEN PRINT CHR$ (43+2*(cc<0));: PRINT ABS cc
6590 PRINT AT 9,2;"-Verticales": IF z4=0 THEN PRINT "NO HAY": GO TO 6610
6600 FOR i=1 TO z4: PRINT d$;"=";e(i+1+z1+z2+z3);TAB 15;: NEXT i
6610 RETURN
6650 CLS : PRINT "PARIDAD E IMPARIDAD": IF NOT coc THEN LET l$="1"
6660 LET x=PI: LET y1=VAL f$/VAL l$: LET x=-PI: LET y2=VAL f$/VAL l$
6680 IF ABS (y1-y2)<1e-6 THEN PRINT "f(";d$;")=f(-";d$;"), luego f es par": GO TO 6710
6690 IF ABS (y1+y2)<1e-6 THEN PRINT "f(";d$;")=-f(-";d$;"), luego f es impar": GO TO 6710
6700 PRINT "La funcion no es ni par ni impar"
6710 PRINT TAB 9;"(compruebalo)": LET x=0: PRINT "Corte con el eje vertical": TAB 9;"f(0)=";
6720 IF VAL l$=0 THEN PRINT CHR$ 154: RETURN
6730 PRINT VAL f$/VAL l$: RETURN
6760 LET a$="("+f$+")/("+l$+")/x": FOR i=0 TO 255: BEEP .01,20
6770 LET x=(i-128)*max/85.3: IF x<>0 AND VAL l$<>0 THEN LET y=(i-128)*VAL a$+88: IF y>=0 AND y<=175 THEN PLOT i,y
6780 NEXT i: RETURN
6790 DEF FN z(x)=VAL f$/(VAL l$+(VAL l$=0)*1e-3)
6810 PLOT 0,87: DRAW 255,0: PLOT 127,0: DRAW 0,175
6820 LET ve=10↑(INT (LN max/LN 7.33333)): FOR n=INT (-3*max/2/ve+1) TO 3*max/2/ve: PLOT FN x(n*ve),90: DRAW 0,-4: NEXT n
6830 PRINT AT 20,21;"UNIDAD" AT 21,21;"ESCALA ";ve: RETURN
6900 RESTORE 6900: DIM h(5,2): FOR i=1 TO 4: FOR j=1 TO 2: READ a$: LET h(i,j)=VAL a$: NEXT j: NEXT i
6910 DATA "3*max/2","bb*3*max/2+cc","(33*max/32-cc)/bb","33*max/32"
6920 DATA "-3*max/2","-bb*3*max/2+cc","(-33*max/32-cc)/bb","-33*max/32"
6930 RESTORE 4560: FOR k=1 TO 3: FOR i=1 TO 4-k: IF h(i,1)>h(i+1,1) THEN FOR j=1 TO 2: LET h(5,j)=h(i,j): LET h(i,j)=h(i+1,j): LET h(i+1,j)=h(5,j): NEXT j
6940 NEXT i: NEXT k: LET x1=FN x(h(2,1)): LET y1=FN y(h(2,2)): LET x2=FN x(h(3,1)): LET y2=FN y(h(3,2)): RETURN
6950 IF VAL l$=0 THEN PRINT CHR$ 154;: GO TO 6965
6960 PRINT FN a();
6965 PRINT TAB 19;
6970 IF VAL m$=0 THEN PRINT CHR$ 154;: GO TO 6985
6980 PRINT FN b();
6985 PRINT TAB 26;
6990 IF VAL n$=0 THEN PRINT CHR$ 154: RETURN
6995 PRINT FN c(): RETURN
9999 LET var=PEEK 23627+256*PEEK 23628: POKE var-6,95+6*i+j: LET n$=j$: RETURN
```

CAMELOT WARRIORS

LA TENSION CRECE.
LA ADRENALINA FLUYE.
LA AVENTURA COMIENZA...
El destino marcó sus leyes en una espada. Cuatro elementos del siglo XX aparecen en un mundo que no es el suyo. La orden de los Dioses fue tajante: Búsqueda, localización y destrucción. Es la lucha de un mundo por su supervivencia. Si apuestas por el riesgo, NO DESESPERES, AQUI ESTA CAMELOT WARRIORS ... y que la fuerza te acompañe.

OPERACION PUZZLE

DINAMIC REGALA 6 MILLONES DE PESETAS EN PROGRAMAS
Más información en los originales



«MANSION DINAMIC»
[illegible] 2, 21, MONTEPRINCIPE
Boadilla del Monte. MADRID

DINAMIC

Tiendas y distribuidores	Pedidos contra reembolso
(91) 447 34 10	(91) 715 00 67



INTERFACE BETA PARA FLOPPY DISK

Rafael PRADES

GRABACION DE FICHEROS

Se realiza de forma similar a la grabación de programas en cinta de cassette, utilizando el comando "SAVE".

Al grabar un programa en código máquina, podemos introducir un tercer parámetro que es opcional, y que hace referencia a la dirección de ejecución. Esta facilidad es muy interesante, ya que permite posteriormente, recuperar y ejecutar un fichero de este tipo, sin tener que confeccionar un pequeño programa cargador en BASIC.

> Ejemplo:
>
> A>SAVE "handler" CODE 40000,756,40200
>
> Direc. de comienzo: 40000
> Longitud: 756 bytes
> Direc. de ejecución: 40200

Es necesario indicar el tipo de fichero, si éste no es BASIC. Ejemplo:

```
A>SAVE "disk"
A>SAVE "game" LINE 250
A>SAVE "oca" CODE
   30000,256
A>SAVE "xray" CODE
   45000,200,45100
A>SAVE "kaos" DATA z()
A>SAVE "golding" DATA
   t$()
```

En el directorio se almacenan los datos correspondientes al nombre, en sectores, etc.

RECUPERACION DE FICHEROS

Podemos cargar y/o ejecutar cualquier fichero, con ayuda de los siguientes comandos:

> LOAD
> RUN
> MERGE

El primero, carga en memoria el fichero especificado. Si éste era BASIC, una vez finalizada la carga, retorna al sistema operativo del Spectrum y visualiza el listado. Los programas en código máquina pueden reubicarse en una posición diferente. Si no hay suficiente memoria, aparece el mensaje:

> • ERROR •

Ejemplos:

```
A>LOAD "gate"
A>LOAD "shift" CODE
A>LOAD "galac" CODE
   34832
A>LOAD "runo" a()
A>LOAD "dat.7" a$()
```

El comando "RUN" permite cargar en memoria cualquier fichero BASIC o código máquina y posteriormente ejecutarlo. Si no se especificó el número de línea de autoejecución (LINE) de un programa BASIC, al salvarlo, empezará a partir de la uno.

Ejemplos:

```
A>RUN "draw"
A>RUN "plot" CODE
```

Para combinar varios ficheros BASIC, se utiliza el comando "MERGE".

Ejemplos:

```
A>MERGE "crash"
A>MERGE "iva.dat"
```

ACCESO ALEATORIO

Con los comandos "PEEK" y "POKE" podemos acceder a cualquier sector de un fichero en código máquina, bien sea en lectura o escritura respectivamente.

Si pretendemos acceder a un sector superior a la longitud del fichero, aparece el mensaje:

> Rec. No. O/F

Ejemplos:

> A>PEEK "display" CODE 34000,2

Almacena a partir de la dirección 34000, los 256 bytes pertenecientes al sector "2" del fichero "display".

> A>POKE "shunt" CODE 43500,5

Graba en el sector "5" del fichero "shunt" el contenido de los 256 bytes posteriores a la dirección 43500.

RESUMEN DE COMANDOS

Selección de la unidad

•"A:"	Selecciona la unidad "A".
•"B:"	Selecciona la unidad "B".
•"C:"	Selecciona la unidad "C".
•"D:"	Selecciona la unidad "D".

Entrada/salida del D.O.S.

RANDOMIZE USR 15360	Entrada al D.O.S.
RETURN	Entrada al S.O. del Spectrum.

Manejo de ficheros

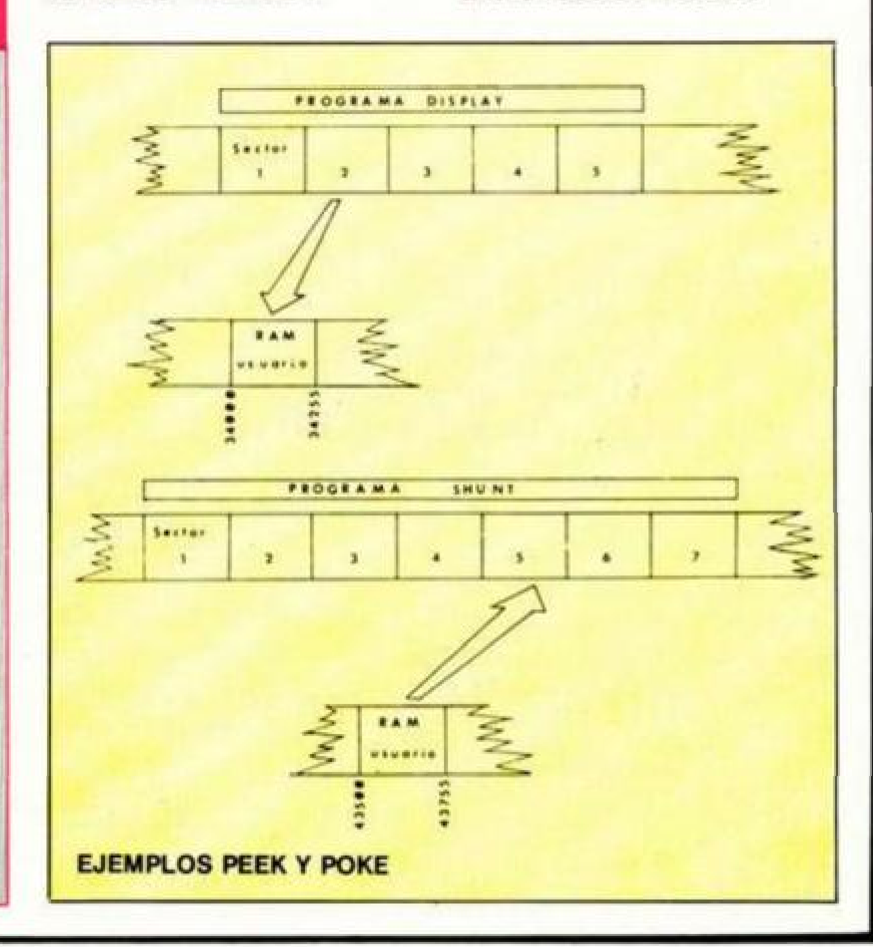
CAT	Visualización del directorio.
ERASE	Borrado de ficheros.
MOVE	Reorganización de ficheros.
NEW	Cambiar el nombre de fichero.
USR	Cambiar la password.

Grabación y recuperación

SAVE	Almacenamiento en diskette.
LOAD	Carga de ficheros.
MERGE	Combinación de ficheros.
RUN	Carga y ejecución.

Acceso aleatorio

PEEK	Leer un sector.
POKE	Escribir un sector.



EJEMPLOS PEEK Y POKE

LOS OPERADORES LOGICOS

Jesús ALONSO

Además de las operaciones aritméticas, logarítmicas y trigonométricas, nuestro ordenador es capaz de realizar una serie de operaciones lógicas pertenecientes a la denominada «Algebra de Boole».
Un perfecto conocimiento de estos «operadores lógicos» puede servirnos para mejorar nuestros programas.

En lenguaje matemático, se denomina Algebra de Boole a «un sistema de entes abstractos cualquiera en que se han definido las operaciones binarias "unión" e "intersección" y la operación unitaria "negación", respecto de las cuales el sistema es cerrado, y que cumplen las propiedades conmutativa, distributiva, de complementaridad y de identidad» (Sixto Rios, «Matemática Finita», Ed. Paraninfo, 1974).

En nuestro caso, trabajaremos con condiciones que puedan ser «verdaderas» o «falsas» y a las que asignaremos, respectivamente, los valores «1» y «Ø».

Si ejecutamos la sentencia "LET a=(b>c)", le estaremos diciendo al ordenador: «comprueba si "b" es mayor que "c"» el ordenador asignará a la variable «a» el valor «1» si la condición "b>c" (b mayor que c) es verdadera, y «Ø» si es falsa.

Las operaciones que vamos a definir se llamarán "OR", "AND", y "NOT", que equivaldrán respectivamente a «unión», «intrersección» y «negación».

Podemos ejecutar: "LET a=(b>c) AND (d>e)" y en este caso, «a» sólo llegará a valer «1» si ambas condiciones son ciertas y valdrá «Ø» en cualquier otro caso.

Si ejecutamos: "LET a=(b>c) OR (d>e)" ahora, será suficiente con que una sola de las condiciones sea cierta para que «a» valga «1», y será necesario que ambas sean falsas para que llegue a valer «Ø».

Por último, si escribimos: "LET a=NOT (b>c)", esta vez «a» valdrá «Ø» cuando la condición sea verdadera, y «1» cuando sea falsa.

Estos resultados se pueden resumir en las denominadas «Tablas de verdad»:

OPERADORES

"AND"	"OR"	"NOT"
Ø AND Ø = Ø	Ø OR Ø = Ø	NOT Ø = 1
Ø AND 1 = Ø	Ø OR 1 = 1	NOT 1 = Ø
1 AND Ø = Ø	1 OR Ø = 1	
1 AND 1 = 1	1 OR 1 = 1	

Los operadores lógicos en el Spectrum

Como una particularidad del Spectrum, podemos notar que el ordenador considerará, sólo a efectos de operaciones lógicas, que vale «1» todo aquello que no valga «Ø». De forma que: "LET a=NOT 3" asignará a «a» el valor «Ø», y "LET a=NOT Ø" asignará a «a» el valor «1».

En una expresión se pueden poner tantos operadores lógicos como se desee. Es perfectamente lícito escribir: "LET a=b AND NOT c OR d" que asignará a «a» el valor «1» siempre que «d» sea «1» (basta con que no sea cero) o «b» sea «1» y «c» sea «Ø», y le asignará el valor «Ø» en cualquier otro caso.

El operador «EXOR»

Supongamos que escribimos:

"LET a=(b AND NOT c OR NOT b AND c)"

que asignará a «a» el valor «1» solamente si «b» o «c» son distintos de «Ø», y asignará a «a» el valor «Ø» tanto si «b» y «c» son cero, como si ambos son distintos de cero.

Esta topología es tan útil que tiene un nombre propio, se conoce como «EXOR» (OR exclusivo) y su «tabla de verdad» es:

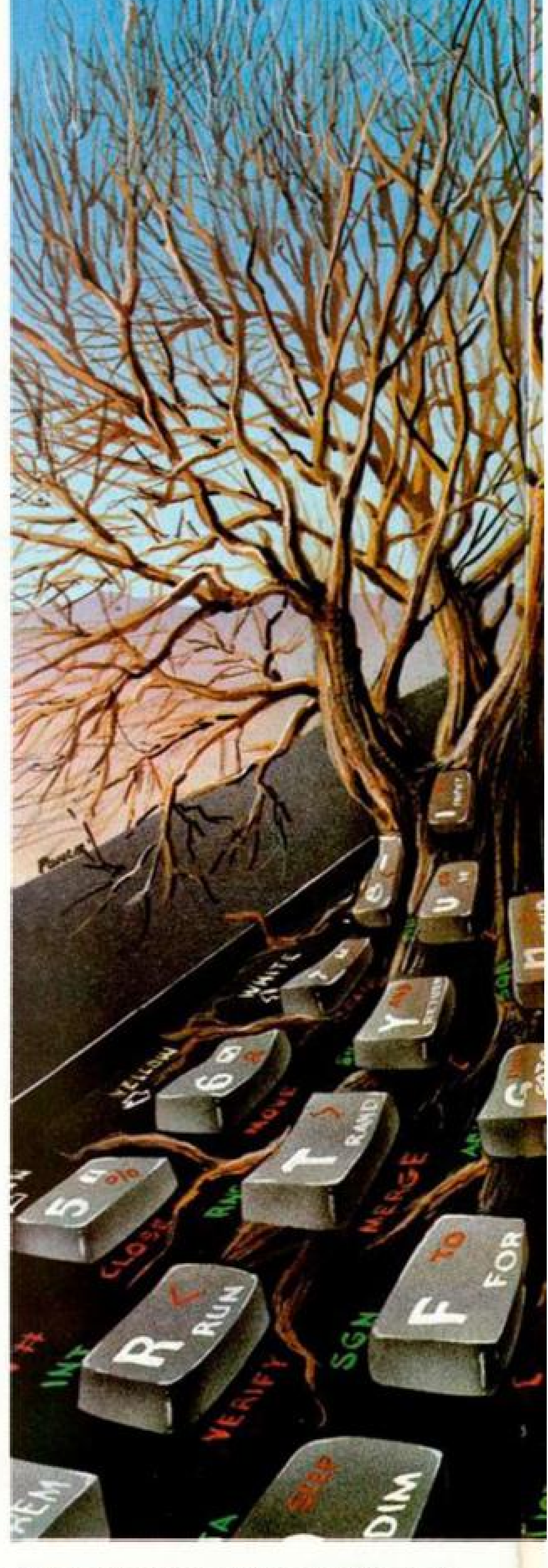
OPERADOR "EXOR"

Ø EXOR Ø = Ø
Ø EXOR 1 = 1
1 EXOR Ø = 1
1 EXOR 1 = Ø

Por desgracia, el Basic no admite la introducción directa del operador «EXOR» (el código máquina, sí), pero podrá construirlo cada vez que lo necesite, de la forma indicada.

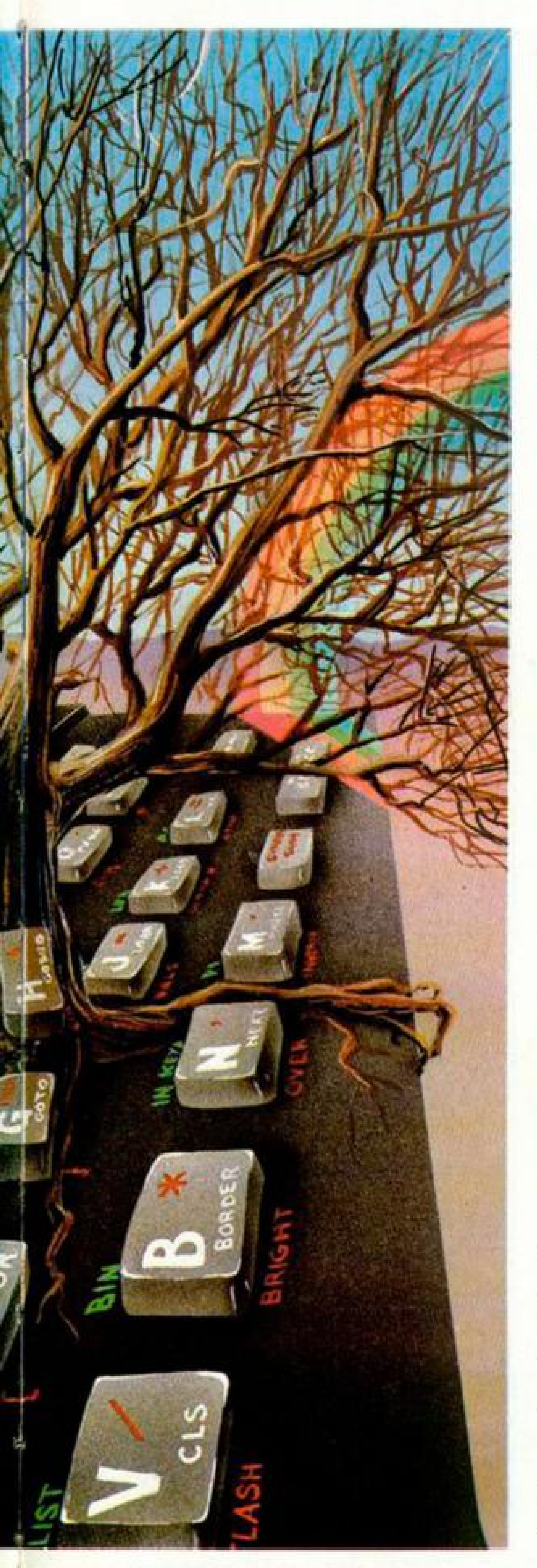
Utilidad de los operadores lógicos

La utilidad más inmediata de los operadores lógicos se encuentra en las sentencias del tipo: "IF... THEN". En



estas sentencias, entre el «IF» y el «THEN» podemos colocar cualquier expresión, el intérprete de Basic valorará esta expresión, y ejecutará lo que sigue a «THEN» sólo si el resultado es distinto de «Ø».

Habitualmente, solemos ver entre «IF» y «THEN» una comparación, que como sabemos, da «Ø» si es falsa y «1» si es verdadera. Pero esta no es la única posibilidad. Las siguientes construcciones son igualmente lícitas:



"IF a THEN GO TO 1ØØ" (El programa salta a la línea 1ØØ si «a» es distinto de «Ø»).

"IF NOT a THEN GO TO 1ØØ" (Opuesto al caso anterior).

"IF a*b THEN GO TO 1ØØ" (El programa salta a la línea 1ØØ si «a» y «b» son distintos de «Ø»).

"IF a AND b THEN GO TO 1ØØ" (Igual que en el caso anterior).

"IF a OR b THEN GO TO 1ØØ" (El programa salta a la línea 1ØØ si «a» o «b» es alguno distinto de «Ø», y también si ambos lo son).

"IF a+b THEN GO TO 1ØØ" (Igual que el caso anterior, excepto que «a» y «b» sean iguales y de signo contrario).

Estos son sólo algunos ejemplos. En general, entre «IF» y «THEN» se puede colocar cualquier expresión valorable por el Spectrum.

Simplificando programas

Es muy frecuente ver en los programas, rutinas del tipo:

```
 100 LET a=CODE INKEY$
 110 IF (a=81) OR (a=113) THEN G
O TO 200
 120 IF (a=65) OR (a=97) THEN GO
 TO 300
 130 IF (a=79) OR (a=111) THEN G
O TO 400
 140 IF (a=80) OR (a=112) THEN G
O TO 500
 150 GO TO 100
```

Se podría tratar de la conocida rutina para mover algo por la pantalla en respuesta a las teclas «Q», «A», «O» y «P».

Si utilizamos operadores lógicos, la rutina anterior quedaría reducida a:

```
 100 LET a=CODE INKEY$
 110 GO TO 100+100*(a=81 OR a=11
3)+200*(a=65 OR a=97)+300*(a=79
OR a=111)+400*(a=80 OR a=112)
```

Puedes constatar que esta rutina funciona exactamente igual que la anterior, con un considerable ahorro de memoria, y una mayor velocidad de ejecución.

En realidad, el hecho de poder mezclar en una misma expresión, comparadores, operadores aritméticos y operadores lógicos, nos permite prescindir de casi todas las sentencias "IF...THEN".

Efectos más asombrosos se consiguen si utilizas operadores lógicos en conjunción con cadenas, aunque en este caso, sólo podrás utilizar el operador «AND».

Supongamos el siguiente programa:

```
  10 INPUT "Numero entre 1 y 22
?";a: CLS
  20 FOR n=1 TO 22
  30 PRINT n;: IF a=n THEN PRINT
 " < Este es su numero": GO TO 5
0
  40 PRINT
  50 NEXT n
  60 GO TO 10
```

Podríamos evitarnos la sentencia "IF...THEN" utilizando el operador lógico «AND» de la siguiente forma:

```
  10 INPUT "Numero entre 1 y 22
?";a: CLS
  20 FOR n=1 TO 22
  30 PRINT n;" < Este es su nume
ro" AND a=n
  40 NEXT n
  50 GO TO 10
```

Comprobarás que ambos programas funcionan de la misma forma, pero en el segundo, nos hemos ahorrado la sentencia "IF...THEN", y con ella, unos cuantos bytes.

Considera ahora el siguiente programa:

```
  10 INPUT "Valor de a ? (entre
1 y 5) ";a
  20 IF a=1 THEN PRINT "a=1"
  30 IF a=2 THEN PRINT "a=2"
  40 IF a=3 THEN PRINT "a=3"
  50 IF a=4 THEN PRINT "a=4"
  60 IF a=5 THEN PRINT "a=5"
  70 GO TO 10
```

No parece una forma muy racional de programar. Lo primero que se nos ocurre es utilizar un bucle "FOR ... NEXT".

```
  10 INPUT "Valor de a ? (entre
1 y 5) ";a
  20 FOR n=1 TO 5
  30 IF a=n THEN PRINT "a=";n
  40 NEXT n
  50 GO TO 10
```

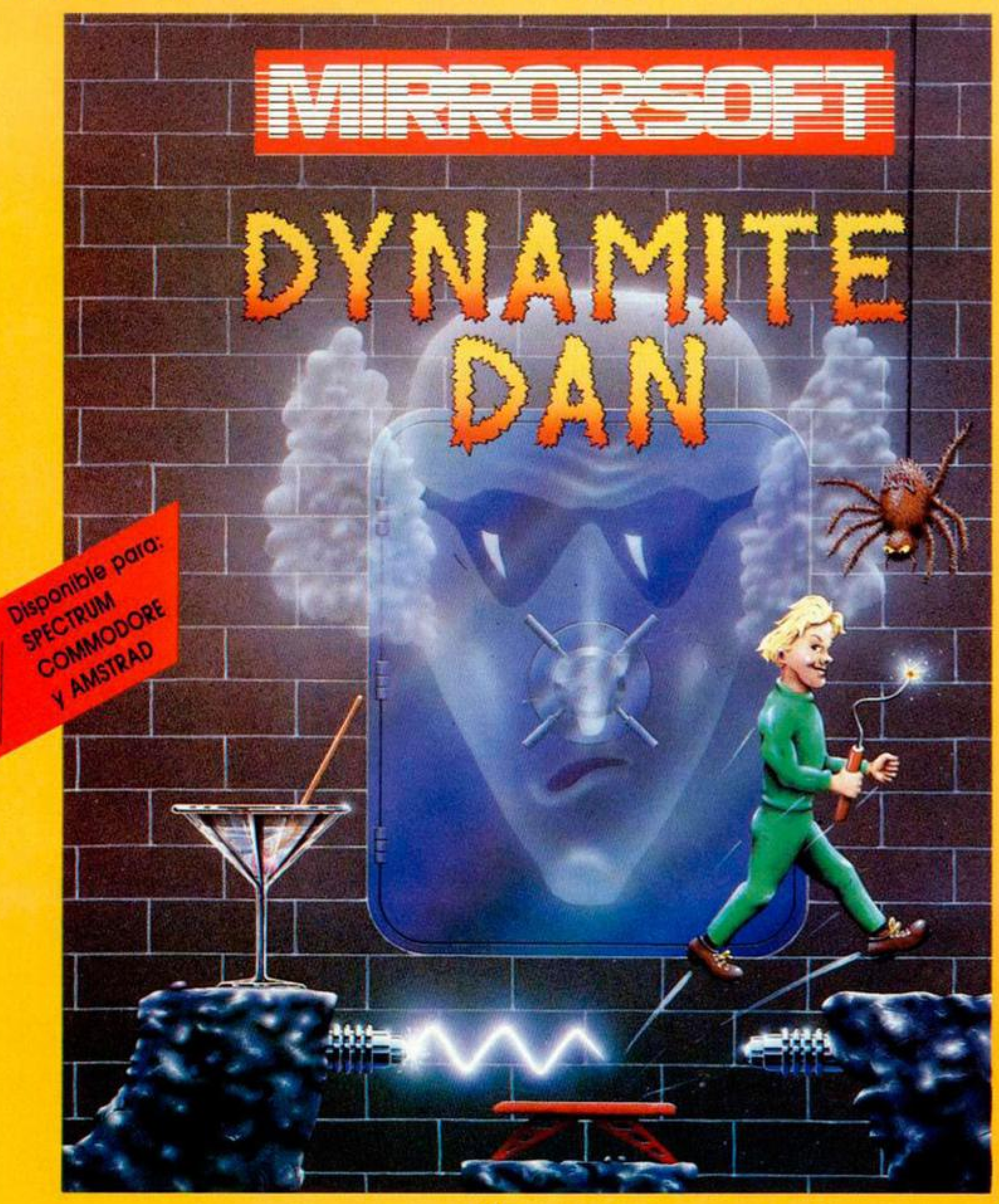
Para la mayoría de los programadores en Basic, esta sería la forma más correcta, pero tiene el inconveniente de que los bucles "FOR ... NEXT" son lentos de ejecutar. Prueba el siguiente programa, aunque no te lo creas, realiza exactamente la misma función, pero mucho más deprisa.

```
  10 INPUT "Valor de a ? (entre
1 y 5) ";a
  20 PRINT "a=1" AND a=1;"a=2" A
ND a=2;"a=3" AND a=3;"a=4" AND a
=4;"a=5" AND a=5
  30 GO TO 10
```

Como verás, la utilización de operadores lógicos puede cambiar completamente la fisonomía de tus programas, no en vano, el funcionamiento de todos los ordenadores se basa en el Algebra de Boole.

Pon DINAMITA a tu imaginación

¡¡POR PRIMERA VEZ EN EL MUNDO UN PROGRAMA DE JUEGOS QUE PUEDES HACER VARIAR A TU MEDIDA CUANTAS VECES QUIERAS!!.



¡No te lo pierdas!

DISTRIBUIDO EN ESPAÑA POR:

círculo de soft

MICROAMIGO S.A.

P.º de la Castellana, 268, 3.º C. 28046-MADRID.
Tel.: (91) 733 25 00

MICRO Manía

Sólo para adictos

CEDIDO POR MICRO Manía

Os ofrecemos en esta sección, cedida por Micromanía, todo tipo de pokes y trucos que os ayudarán a conseguir mejores puntuaciones en los juegos, recordándoos que estamos abiertos a cualquier tipo de colaboración por vuestra parte. Para ello no tenéis más que enviar vuestros descubrimientos y aportaciones a MICROHOBBY, poniendo en el sobre «SECCION MICROMANIA».

PSYTRON

Si tenéis olvidado en el fondo del cajón este antiguo pero excelente juego, os ofrecemos una buena ocasión para que le quitéis un poco el polvo y volváis a disfrutar con él, pero en esta ocasión con algunas sustanciales ventajas.

Para introducir los pokes, deberéis actuar del modo usual: teclear MERGE"" y cuando aparezca O.K. parar la cinta. Después podréis introducir los siguientes Pokes:

POKE 28625,0:POKE 28626,0 para fuel inagotable.

POKE 41098,17:POKE 41099,2:POKE 41100,1:POKE 41101,0 para vidas infinitas.

SPY HUNTER

Un pequeño truco para este magnífico juego de U.S. Gold. Si vas siguiendo al camión que aparece al principio, y tienes cuidado de no echarle fuera de la carretera o de disparar contra él, no aparecerá ningún coche de agentes enemigos en la pantalla.



MUGSY

Ojalá que conseguir dinero en la vida real fuera tan sencillo como hacerlo en este programa. Para hacerse con millones de dólares, lo único que debes hacer es, cuando te pregunten que cuánto deseas gastarte en artillería y munición, teclear 999999999 (es decir, nueve nueves) y podrás hacerte millonario de la manera más tonta. Pero ten cuidado, si vuelves a teclearlo, el juego se colgará.

Fallos en el teclado del Plus

Tengo un Spectrum Plus y desde los primeros días, de una forma esporádica y sin coincidencia con alguna manipulación determinada, el aparato presenta los siguientes fallos:

— No borra (DELETE); si quiero hacerlo, tengo que mantener pulsado «DELETE» y borrar con el «Ø».

— No funciona el «Cursor a derecha», esta función tengo que hacerla manteniendo pulsada dicha tecla y, simultáneamente, el «8».

— El «;» me sale únicamente si mantengo pulsada «;», «"» o «SYMBOL SHIFT» indistintamente y la «o»; para que me salgan «"» lo mismo, pero pulsando la «p».

Y otra serie de fallos similares en las teclas «GRAPH.», «CURSOR ARRIBA», etc.

El teclado lo desmonté y comprobé con el ohmetro que los contactos son buenos. ¿Cuál puede ser la causa y qué solución tiene?

F. de LUIS - Barcelona

☐ El teclado del Spectrum Plus presenta, en ocasiones, este tipo de problemas, que se producen siempre en las teclas especiales, es decir, aquéllas que no existían en el teclado original del Spectrum, y que se deben a un mal contacto en la triple membrana.

La solución más recomendable consiste en sustituir el teclado por otro de mayor calidad, pero esto implica un desembolso de dinero que no todos los usuarios están dispuestos a realizar. Por nuestra parte, hemos pasado el problema a la sección de hardware, donde nuestros expertos están trabajando sobre una posible solución.

De momento, hemos encontrado un sistema bastante sencillo de resolver el problema en la mayoría de los casos. Consiste en desmontar el teclado y colocar trocitos de papel (por ejemplo, confetis) debajo de la membrana, y justo en los lugares correspondientes a las teclas que fallen. La finalidad de esta «chapuza» es que, al apretar una tecla, la membrana sea comprimida con más fuerza y se asegure el contacto. A pesar de lo «etrusco» del sistema, lo cierto es que funciona bastante bien. Si decide intentarlo, no deje de contarnos el resultado.

Protección de software

¿Hay algún modo para que no se pueda ver el listado de un programa, pero que se pueda ejecutar ese programa? Ya vi un sistema en el número 45 de su revista, pero lo único que hacía era poner el papel de negro.

Estoy haciendo un programa en el que hay que bajar escaleras. Quisiera saber qué hay que hacer con «SCREEN$» o «ATTR» para que cada vez que acabe un escalón, baje al próximo.

Aritz ARANBURO - San Sebastián

☐ El sistema que nos indica salió en el apartado «TRUCOS» y lo que hace es apuntar 255 en la variable del sistema que controla la amplitud del canal 1 (pantalla inferior).

Existen múltiples sistemas para evitar que se pueda acceder al listado de un programa, aunque casi todos tienden a evitar que el programa pueda ser detenido (Anti-BREAK) o a conseguir que el ordenador se «cuelgue» si esto ocurre. Mucho de estos sistemas han sido expuestos en nuestra sección «TRUCOS».

Otro sistema que permite detener el programa pero impide ver el listado, consiste en colocar al principio de la primera línea un código de control de color que obligue a que el listado salga con la tinta del mismo color que el papel, con lo cual el listado estará allí pero será imposible verlo.

En cuanto a las escaleras, es bastante sencillo; basta con detectar que no hay «suelo» bajo el muñeco y hacerle bajar. Suponemos que generará la escalera a base de «UDGs», por lo que la función «SCREEN$» no le servirá para nada; utilizaremos «ATTR». Para ello, la escalera deberá tener distintos atributos que el resto de la pantalla; supongamos que la pantalla tiene tinta

negra y papel blanco (ATTR =7 • 8 + Ø=56) y la escalera tiene tinta roja (ATTR=7 • 8 + 2 = 58); si la escalera desciende hacia la derecha y las coordenadas del muñeco vienen dadas por «li» para la línea y «co» para la columna, cada vez que avance el muñeco hacia la derecha (LET co=co+1) deberá hacer la siguiente comprobación:

```
IF ATTR (li+1,co) <> 58
THEN LET li=li+1
```

Antes de imprimir la nueva posición del muñeco con: PRINT AT li,co;... (se supone que la antigua posición ha sido ya borrada). Para subir la escalera, la comprobación habría que hacerla cada vez que avanzara el muñeco hacia la izquierda (LET co=co−1) y sería:

```
IF ATTR (li,co)=58 THEN
LET li=li−1
```

Con lo que el muñeco sería capaz, también de subir la escalera.

Versión

Según he oído, existen varias versiones del Sistema Operativo del QL, etiquetadas con las letras FB, AH, JM, JS. ¿Cómo se puede saber qué Sistema Operativo tiene un QL determinado? ¿Tiene previsto Sinclair lanzar próximamente una nueva versión?

Francisco MARTINEZ - León

□ Efectivamente, existen varias versiones del S.O. del QL; puede saber cuál es la suya si lee el contenido de la variable «ver$», por ejemplo:

```
PRINT ver$
```

Por otro lado, parece que Sir Clive está preparando un nuevo «QL II», es poco lo que se sabe aún de la máquina, pero les aseguramos que cuando nos llegue más información, nuestros lectores serán los primeros en saberlo.

Problemas con el Microdrive

Cuando hago SAVE hacia el Microdrive tengo problemas a la hora de verificarlos, ya que la cinta no para con el «O.K.» ni el fichero se puede cargar, a pesar de aparecer en el catálogo.

Jesús IRIONDO - Vizcaya

□ Se trata del problema más frecuente al trabajar con Microdrive. Puede guardarse el archivo, pero a la hora de verificarlo, el microdrive se detiene al cabo de casi 2 minutos con el informe: «File not found» (no es que no se pare, es que tarda mucho en pararse) a pesar de que el archivo figura en el catálogo.

Lo adecuado en estos casos es borrar el archivo con ERRASE y volverlo a guardar; si sigue teniendo problemas, es que algún sector se ha «machacado» después de haber sido formateado el cartucho. Lo mejor entonces es copiar todos los archivos de ese cartucho en otro y volverlo a formatear.

RAM

Quiero hacer algunas preguntas:

— Diferencia entre RAM dinámica y RAM estática.

— ¿Hay alguna forma de modificar el Hardware para conseguir más colores?; por ejemplo, el bit de brillo que sirva para hacer otros 8 colores de tinta y prescindir del brillo. ¿Se podría hacer sin cambiar la ULA y colocándole algún circuito adicional? ¿O no ha hecho aún nadie nada de esto?

Vicente VICENTE - Valencia

□ Básicamente, la diferencia entre RAM dinámica y estática es que la segunda mantiene la información mientras esté alimentada; en cambio, la primera sólo la mantiene durante un breve lapso de tiempo (de forma similar a un condensador) y es necesario estar continuamente leyéndola y volviéndola a escribir, lo que se denomina: «refresco de memoria». Por otro lado, la RAM dinámica es, considerablemente, más barata que la estática.

Es imposible aumentar el número de colores sin cambiar tanto la ULA, como gran parte del Sistema Operativo; lo que equivaldría a diseñar el ordenador de nuevo.

DE OCASION

● VENDO Impresora Seikosha GP-50S, nueva, ideal para Spectrum, con 18 rollos de papel para ella, por sólo 19.000 ptas. Interesados llamar al Tel. (91)7471034. Preguntar por Andrés.

● VENDO Spectrum 48K, con garantía Investrónica, instrucciones en castellano, cinta de demostración, libros de J. Sales Roig sobre Código Máquina, y tres revistas de «ZX». También compro las instrucciones del procesador de textos para Spectrum 48K «Context». Interesados dirigirse a Frances Martínez Valdés. Vila Mari, 33, Pral-1. Barcelona. Tel. (93)2241113 (20 a 23 h).

● DESEO contactar con estudiantes de ingeniería o arquitectura que sean usuarios de Spectrum para el intercambio de ideas, listados, etc., con fin dirigido a los estudios antes citados. Marcos Alvarez Castro. Ctra. Vizcaína, 45, 7.° B. Gijón. Tel. (985)346810.

● VENDO Amstrad CPC-664 con monitor en color, 2 discos originales con cp/m, logo, base de datos, etc., y manual en Español. Garantía oficial Amstrad España a estrenas. Todo por 110.000 ptas. Tel. (91)2067521.

● VENDO Interface Joystick programable (Indescomp). Nuevo por 4.000 ptas. Precio real: 5.250 ptas. Permite controlar con joystick cualquier juego. Interesados llamar al Tel. 4573720 (21 a 22 h). Preguntar por José. Madrid.

● VENDO consola de video-juegos Philips G-7000 en perfecto estado por 20.000 ptas. Incluye dos cartuchos valorados en 7.000 ptas. Interesados llamar el Tel. 4468788. Preguntar por Alberto. Madrid.

● VENDO O CAMBIO aplicaciones para Sinclair QL (fortram, pascal, basic). Interesados escribir a la siguiente dirección: José M.ª Bracho. Cami Real, 61, 2.° C. Valencia. Tel. 2664279.

● VENDO Videopac Philips G-7000 en perfecto estado con 8 cartuchos de juegos. Precio a convenir. Jordi. Tel. (93)2112327.

● REGALO a quien me compre Spectum Plus, comprado en julio con garantía por un año, manual en castellano, cinta de demostración, alimentación, cables. Les regalo joystick Kempton con interface, 30 revistas Microhobby, todo nuevo. También vendo TV, color por 15.000 ptas. Todo por 18.000 ptas. Llamar al Tel. 2612515 de Madrid.

● INTERCAMBIO ideas, trucos, información, etc., relacionado con el Spectrum 48K. Interesados escribir a F. López. Plaza Fuensanta, 2. Murcia.

● DESEO contactar con usuarios del Spectrum de Valencia y región, para el intercambio de ideas, información, etc. Escribir a Toni. Gregorio Gea, 7, 1.ª Valencia.

● VENDO ordenador Spectrum 48K, en perfecto estado, con todas sus conexiones. Precio: 22.500 ptas. Escribir a José Zurdo. Ledesma, 7-9, 4.° A. Salamanca o bien llamar al Tel. (923)245769 (a partir de las 7).

● HA NACIDO un club de Spectrum, para intercambio de información, escribe a la siguiente dirección: Agustín Luis Estevez. Urb. Tamarco, 60. Tegueste (Tenerife).

● VENDO ZX Microdrive más Interface 1, regalo cartuchos, 2 libros sobre el Microdrive por sólo 20.500 ptas. También vendo sintetizador de voz para Spectrum (Micro-Speech), con amplificador de sonido a través del TV. Sólo por 7.000 ptas. Llamar a Luis Fernando al Tel. 2613627 de Madrid.

● VENDO ZX Spectrum 48K (comprado en abril de 1984), funcinando perfectamente por sólo 16.000 ptas. También teclado profesional DkTronic nuevo por 4.000 ptas. Interesados llamar al Tel. (93)3454202.

● NO TIRES tus cartuchos de Microdrive atascados o con la cinta estropeada. Los compro todos, si están sin abrir. Pago 150 ptas./unidad o 100 ptas. (si no tienen funda). Enviar a Javier Belenguer. Artes Gráficas, 16. Valencia. Tel. (96)3603968.

● VENDO el siguiente lote: ZX Spectrum 48K, TV b/n Sanyo 12''; Interface tipo Kempton y joystick; 8 números de la revista «ZX»; 3 números de Micromanía, todos los n.° aparecidos de la revista «Run» también de Microhobby, 3 libros de programación del ZX Spectrum. Precio a convenir. Para mayor información llamar al Tel. (93)3863986. Preguntar por Epi.

● VENDO vídeo-juegos Atari-2600 con 1 joystick, accesorios e instrucciones en castellano, con 5 cartuchos. Todo en perfecto estado de funcionamiento. Precio: 25.000 ptas. (negociables). Preguntar por Javier. Tel. (968)235114. Murcia.

● VENDO Spectrum 48K, totalmente nuevo, con manuales en castellano, cinta de presentación, por sólo 28.000 ptas. Llamar al Tel. (968)843232, o bien escribir a la siguiente dirección: Jesús M. González Estévez. Avda. García Sánchez, 15, 2.° A. Pontevedra.

● VENDO Impresora Seikosha GP 500A junto con interface y cables para Spectrum con garantía fechada en 28-8-85. Compradores llamar al Tel. (91)4101415, preguntar por Alfonso. También estoy interesado en comprar un Amstrad CPC-664. Pueden escribir a la siguiente dirección: Alfonso de la Fuente. Almagro, 38. Madrid 28010.

● VENDO Spectrum 48K, con cassette especial para ordenadores por 33.000 ptas. Interesados llamar al Tel. (965)283382 o bien escribir a José Luis Santana. Arturo, 9, 3.° D. Alicante.

● VENDO ZX Spectrum Plus, poco usado, con todos los cables, fuente de alimentación, manual, cinta de demostración, 1 amplificador de sonido «Megasound». Tan sólo por 30.000 ptas. Regalo 6 libros de Basic Spectrum con programas y más de 50 revistas. Interesados escribir a la siguiente dirección: Demetrio Irigoien. Goienkalde, 16, Urdain (Navarra).

● VENDO juego de coches con adelantamientos que mide 165 x 70 cm. El precio es de 7.500 ptas. (negociables), o bien lo cambio por joystick e interface tipo Kempston. También intercambio todo tipo de ideas, trucos, etc. Escribir a Guillermo Perucho. La Sequía, 22. Vila Nova de Castelló (Valencia). Tel. (96)2450528.

● ARREGLO teclados originales del ZX Spectrum 16/48K. Para más información llamar al Tel. (942)227447. Preguntar por Jorge. Santander.

● DESEARIA que algún lector me mandase el mapa del juego «Babaliba» de Dinamic o bien pago fotocopias y gastos de envío. Escribir a Oscar Pampím Rodríguez. República Argentina, 7-9, 2.° derecha. El Ferrol (Coruña).

● VENDO Commodore 16, con tres meses de garantía más curso de Basic I, por 25.000 ptas. Escribir a Fco. Javier Arévalo Pérez. Bda. Juan XXIII. Sevilla. Tel. (954)640851.

● VENDO/CAMBIO Spectrum Plus con cables, libros, etc., y con interface programable con joystick (julio-85) y cassette para ordenador. Regalo revistas. Barato, éste cambio es por un Commodore CBM-64K, en condiciones. Llamar al Tel. (91)7060299.

● DESEARIA que algún lector me mandara las instrucciones en castellano del juego «Dragontorc», llegaría a pagar 300 ptas. o lo cambiaría por las instrucciones de otros programas. Los interesados pueden escribir a: Idioma Esperanto, 28, B° C. Madrid 28017.

● VENDO Spectrum Plus, comprado en marzo-85, poco usado, regalo libro «El programador de Spectrum» y revistas por sólo 30.000 ptas. (negociables). Escribir a Venancio Valbuena Rodríguez. Sta. Catalina, 31. Benavente (Zamora).

● CORREOSPECTRUM-MADRID Para todo usuarios de Spectrum que desee formar parte de un club, con revista mensual. Trucos, información, ideas, etc. Escribir a Javier Serrano Sánchez. Apóstol Santiago, 49, 3° C. Madrid 28017.

● URGE vender Spectrum 48K con interface Turbo con salida para los joystick y cartuchos. También incluyo el joystick, revistas, estuches para cassette. Todo por 50.000 ptas. Interesados llamar al Tel. 338659 o escribir a Juan José Granados Luque. Sta. Rosa, 49, 6.° Pta. 24. Alcoy (Alicante).

● VENDO Spectrum 48K, con manuel en español por sólo 23.000 ptas. Interesados: Antonio Suárez. S. Bernardino, 15. Madrid 28015. Tel. 2481461.

● VENDO Commodore Vic-20 (dos meses de uso), con instrucciones en castellano, libro de experiencias, por sólo 20.000 ptas., o bien por un Spectrum 48K. Llamar a J. Luis García. Poniente, 44, Aspe (Alicante). Tel. (965)491137.

AVISO A NUESTROS LECTORES

Comunicamos a todos nuestros lectores que a partir del día 1 de enero de 1986 nuestro Departamento de Clientes será trasladado al Polígono Industrial de Alcobendas. La Granja, 39. Alcobendas (Madrid). Para cualquier consulta que deseen realizar, pueden llamar al Tel.: 654 28 98 - 67 de 9 a 14.30 h y de 16 a 18.30 h.

DISEÑA TUS PROPIAS PANTALLAS; HAZ LA MUSICA QUE TE GUSTE Y DIVIERTETE JUGANDO CON...

Este mes:

YOUR COMPUTER

Te ofrece algo realmente sabroso:

Drawer

El mejor y más completo programa para diseñar pantallas. Los potentes comandos que posee transforman la tarea en un apasionante juego.

- Dos rejillas guía (color y B/N).
- 20 tramas de relleno predefinidas o diseñadas por ti mismo.
- Ampliación de gráficos (8 y 16 aumentos).
- Trazo automático de líneas y círculos.
- Almacén temporal de pantallas.
- 10 velocidades de desplazamiento.
- Modos, texto y gráficos definidos.

Synsound

Transforma tu ordenador en un auténtico sintetizador de sonidos. Podrás componer hasta 7 melodías y ejecutarlas en combinación con cualquiera de los efectos propios del programa o, si lo prefieres, créalos tú mismo.

Aliquid Simplex

Completa tu experimento antes de que el incendio destruya tu laboratorio.

Asteroids Atack

Atravesar el espacio intergaláctico supone un alarde de habilidad propio del máx experimentado piloto. ¿Lo eres tú?

Ballon Hooper

Apropiarse de lo ajeno no suele resultar fácil, su dueño se pondrá «muy pesado».

Blockpaint

Con DRAWER es fácil pintar..., pero si tu monitor se rebela, las cosas se te pondrán muy difíciles.

2 YOUR COMPUTER

EL CORAZON DE LA PRIMERA REVISTA EUROPEA DE ORDENADORES

SPECTRUM 48, PLUS, 128

695.- PTAS.

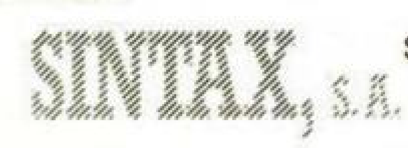
La mejor selección de programas de juegos y utilidades, publicados en la revista de mayor difusión de Europa en ordenadores. Ahora reproducidos en cassette, en auténtica exclusiva mundial.

1 Aliquid Simplex
2 Asteroid Attack
3 Ballon Hooper

¡YA ESTA A LA VENTA EN TU KIOSCO!

¡GANA UN 128 K!

Total garantía de carga



Si no lo encontrara en su kiosco, puede solicitarlo directamente a nuestra editorial:
Paseo de la Castellana, 268. Tel.: (91) 733 25 99. 28046 Madrid.

SPECTRUM 128
EL SUMMUM

Spectrum, como líder, marca un nuevo hito en la historia de los ordenadores familiares.

El Spectrum 128.

Gran capacidad de memoria. Teclado y mensajes en castellano, teclado independiente para operaciones numéricas y de tratamiento de textos...

Sinclair e Investrónica han desarrollado una auténtica novedad. En ningún lugar del mundo, salvo en los Distribuidores Exclusivos de Investrónica, podrás encontrar el nuevo Spectrum 128.

Sé el primero en tener lo último.

SPECTRUM 128. NOVISSIMUS



investronica

Tomás Bretón, 62.
Tel. (91) 467 82 10.
Telex 23399 IYCO E.
28045 Madrid

Camp, 80.
Tels. (93) 211 26 58 - 211 27 54.
08022 Barcelona